ANALYSIS OF EFFECTS OF ORGANIZATIONAL BEHAVIOR ON EVOLVING SYSTEM OF SYSTEMS ACQUISITION PROGRAMS THROUGH AGENT BASED MODELING

THESIS

Klayton S. Bobsein, Captain, USAF

AFIT-ENV-13-M-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENV-13-M-03

ANALYSIS OF EFFECTS OF ORANIZATIONAL BEHAVIOR ON EVOLVING
SYSTEM OF SYSTEMS ACQUISITION PROGRAMS THROUGH AGENT BASED
MODELING

THESIS

Presented to the Faculty

Department of Systems and Engineering Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Systems Engineering

Klayton S. Bobsein, BS

Captain, USAF

March 2013

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

AFIT-ENV-13-M-03

ANALYSIS OF EFFECTS OF ORANIZATIONAL BEHAVIOR ON EVOLVING
SYSTEM OF SYSTEMS ACQUISITION PROGRAMS THROUGH AGENT BASED
MODELING

Klayton S. Bobsein
Captain, USAF

Approved:

_____//Signed//_____   _07 Mar 2013_
John M. Colombi, Ph.D. (Chairman)   Date

_____//Signed//_____   _07 Mar 2013_
John J. Elshaw, Lt Col, USAF (Member)   Date

_____//Signed//_____   _07 Mar 2013_
Joseph R. Wirthlin, Lt Col, USAF (Member)   Date

Abstract

Every weapon system operates in context with one or more system of systems (SoS). Generally, it is the SoS that provides warfighting capability. However, each system is managed independently by a program office with program-centric priorities: requirements, funding and schedule. As needed, these systems must be interconnected and interoperable, so the program office must collaborate across the SoS with other program offices. Thus, the SoS and the constituent systems are always changing and evolving, triggered by users needs, new threats and various stakeholders demands.

Acquisition program offices can be characterized with a set of inherent organizational behaviors that respond to the environment, are influenced by the SoS architecture, and can be described by their fitness and contribution to the SoS. Using Geert Hofstede's cultural dimensions, integrated with a modified version of the Bak-Sneppen biological evolutionary model, this research highlights which set of behaviors are significant in affecting the overall SoS fitness. Through the use of agent based modeling, it was determined that the organizational behaviors of willingness and ability were significant factors to predict local fitness with a correlation of 0.548 and 0.535 respectively. Using these factors with local fitness, a regression model was built to better predict the local fitness of the system. Global fitness was highly dependent on the influence from connected systems, which surprisingly remained highly stable throughout different modeling variations in learning strategies, prior fitness contribution, trigger types, selection percentage, and fitness degradation efforts. This first-of-its-kind research provides a starting point into complex integration of organizational behavior and SoS architecture and their impact on acquiring and delivering warfighter capabilities.

"Be the Fastest and the Best"
To my wife and children

Acknowledgments

I dedicate to my time, effort, and research to my family. Without their support and understanding, my ability to focus on my research could not be achieved. Also, I would to thank Dr. Colombi for his support, patience, and creativity in our discussions to keep me moving forward on this research. His experience and knowledge of MATLAB made it possible for me to implement my ideas into code. Finally, I would thank Lt Col Elshaw for allocating time out of his schedule to accommodate an independent study that formed the basis of this research.

Klayton S. Bobsein

Table of Contents

List of Figures

List of Tables

List of Equations

ANALYSIS OF EFFECTS OF ORANIZATIONAL BEHAVIOR ON EVOLVING SYSTEM OF SYSTEMS ACQUISITION PROGRAM THROUGH AGENT BASED MODELING

I. Introduction

According to Gen Mark A. Welsh III, "The Air Force has five priorities: continuing to strengthen the nuclear enterprise, winning today's fights, developing and caring for Airmen and their families, modernization, and *recapturing acquisition excellence*." (Atkins 2012)

General Issue

Over the course of the last two decades, the United States military has shown that it fights with collections of individual systems, working together to achieve operational objectives. Each of these systems has an acquisition program office, which coordinates their efforts with other program offices to increase overall operational capability to form many System of Systems (SoS).   An SoS may or may not have an overall program manager, chief architect, or system engineer to ensure that some overall performance level is met. Current military acquisition guidelines (DoDI 5000.02) train current and future acquisition officers and leaders with the basic knowledge of handling a single system, not multiple systems like an SoS. Fully understanding the complexities of a large single system takes time to grasp and comprehend, but today's environment of increasingly more SoS's leads one to wonder how one program office can accomplish it and how an SoS with no overarching leadership can meet operational needs. Even with centralized SoS leadership like Missile Defense Agency (MDA), it may still be hard to ensure the required operational performance. As with nature, the operational

requirements and the needs of the warfighter evolve and require the SoS to evolve as well in order to support on-going operations. These new requirements and warfighter needs "trigger" program offices to balance their resources like funding and personnel. With evolving SoS architectures and complex interrelated baselines, it becomes crucial for the SoS to understand the organizational behavior and relationships that affect overall performance of the SoS. In Figure 1 below, this is an example of a physical architecture of an SoS. This shows the individual systems and their interfaces between other systems, but all of these individual systems are not usually controlled by one program office. Figure 2 provides an illustration of the types of organizations involved with each physical system in Figure 1. The dashed lines demonstrate the potentially loss of control or responsibility over a particular set of systems. Also, there are more support organizations that are not represented in the physical architecture of the SoS but are necessary to meet the objectives of the mission.
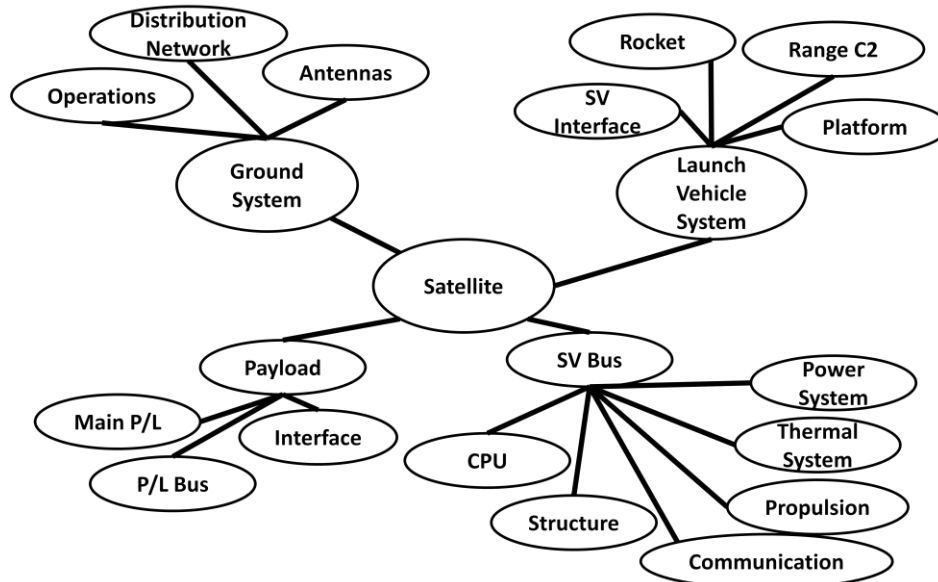
Figure 1: Physical SoS Architecture
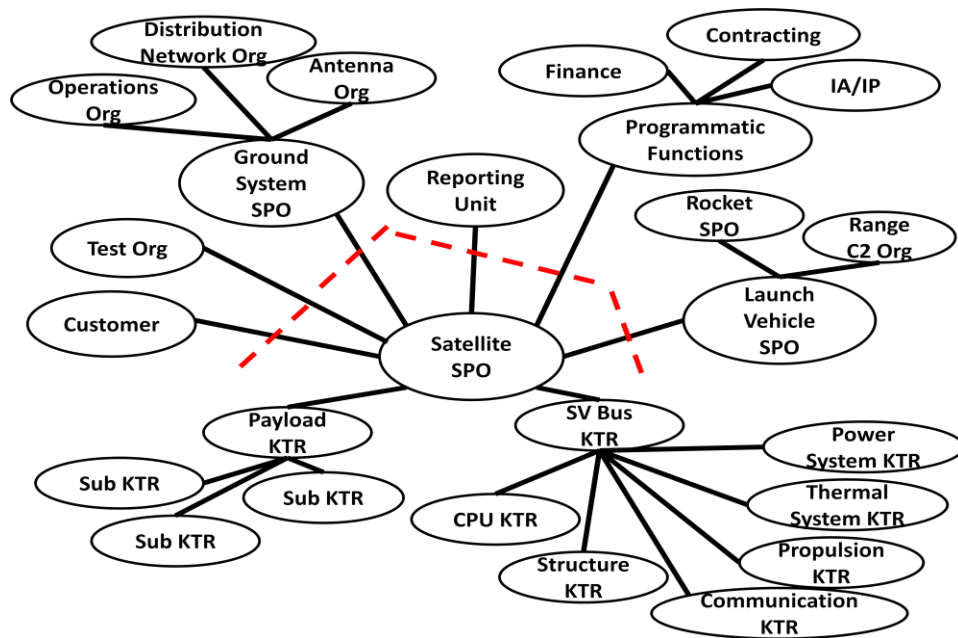
Figure 2: Socio-Organizational Architecture

In a recent presentation at the National Defense Industrial Association (NDIA) Systems Engineering Conference, Ms. Kristen Baldwin and Dr. Judith Dahmann reviewed 47 Major Defense Acquisition Programs at various stages of development and identified three issues: SoS context, management, and technical, as potential areas of concern that could affect capability needs. They identified that many SoS issues focus on the complexities of the program across on multiple, independent organizations. They calculated that almost half of the major defense acquisition programs had management issues, which included lack of formal agreements such as a contract, poorly defined roles and responsibilities, and the "approach to organizational coordination is unclear" (Baldwin and Dahmann 2012). Another key contributor to the difficulty within an SoS is lack of understanding of relationships. Due to a lack of understanding of the complex nature of the SoS, programs could not provide a coordinated effort amongst its individual components during the acquisition of the SoS. This study revealed the need within DoD

3

acquisition programs to understand how organizational behavior and inter-system

relationships affect the acquisition of an SoS.

Problem Statement

When faced with the reality of smaller budgets among a collection of diverse

program offices, the ability to assess the changing performance levels and behavior

across the SoS is not fully understood by program managers or stakeholders. As more

requirements and change requests are being levied on an SoS, the architecture evolves

and the systems are modified to meet these new requirements. This includes updating

and improving a system or re-networking an SoS through communication technology and

collaborating with other program offices to obtain new capabilities. In today's

environment, new capabilities can be added to current systems, but they could end up

degrading the overall performance of the SoS and require a more in-depth analysis to

correct the newly deployed SoS architecture. For the purposes of this research, the

performance of the individuals and the SoS is characterized by maintaining a certain level

of the fitness. Here, it is analogous to biological fitness, where the strength of the system

maintains its survival within an environment. By understanding the environment and

organizational behavior, system engineers can implement a wave model approach to an

SoS in order to provide an on-going analysis of the SoS to recognize how new

requirements affect the behavior of the organizations involved in the SoS, which affects

the overall fitness of the SoS.

Research Objectives/Questions

The objective of this research was to design a simulation to explore how and why changes occur in an SoS. To address the problem statement, this research answers the following questions:

1. What are the triggers that initiate SoS change and how they will impact an SoS's "fitness"?

2. What are the business rules and associated behavior of the involved organizations (SoS program office, individual system program offices, stakeholders, etc.)?

3. How can the system and SoS "fitness" be calculated as a function of its systems' willingness, ability, and architecture?

Hypothesis

The environment surrounding an SoS will provide a stimulus, which the systems within the simulation respond to in order to interact with other systems and the environment. Behaviors and business rules will provide the basis from which the systems can make decisions. These behaviors and interactions amongst the other systems and environment will influence the overall fitness of the SoS.

Methodology

First, a generic architecture was developed to provide a baseline for the simulation and context for the organizational behavior. Once the type of simulation was selected, the development of relationships and the rules of behavior between the systems and environment are established. These rules were based on organizational behavior and group dynamics, researched to better understand how organizations function and

cooperate or hinder each other. The simulation was built based on the criteria above and allowed to run over multiple iterations in order to show the evolution of the SoS. Finally, the overall SoS fitness will be evaluated after each iteration to determine how the stimulus and behaviors are influencing the SoS.

Assumptions/Limitations

Detailed organizational assessment and leadership characteristics were unknown, so the model and architecture was based on a generic set of rules and organizations. The rules and architecture was developed based on past experiences, discussions with subject matter experts and research from other models. With this generic model, the results derived from this research will provide insight into this model alone and does not represent any current programs. Also, not all aspects of the acquisition process will be captured to the nature of this research. The focus will be on several organizational characteristics and SoS fitness.

Implications

This research will provide decision makers and acquisition programs an understanding of how behavior impacts collective fitness, not just individual program schedule, funding, and resources. Also, this will provide policy makers and stakeholders an insight into how a new requirement and other system modifications and triggers may affect the behavior of an SoS.

II. Literature Review

Chapter Overview

As a system and SoS evolve to meet the demands of the warfighter, program offices must evaluate their current resources and balance them to meet the new requirements. First, the term, system, will be defined to establish a baseline for the organizational structure for a generic model, followed by a discussion of the SoS environment as it relates to acquisition. Then, acquisition triggers will be discussed to demonstrate why an SoS changes. Next, an investigation into organizational behavior is required to provide context for the decision making that the individual organizations will have as the simulation is initiated and SoS requirements change. Finally, a review of the different types of simulation tools will be evaluated, and one will be determined the right type of simulation tool needed to construct the proper interaction and complexity of program offices and their corresponding environments.

Acquisition Concepts

To understand the research and the modeling at hand, baseline concepts need to be established to provide the basis for evaluating the evolution of an SoS acquisition. Maier and Rechtin state that "systems are collections of different things that together produce results unachievable by the elements alone" (Maier and Rechtin 2009). Along these lines, the *Systems Engineering Guide for Systems of Systems* defines a system as a "functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements; that group of elements forming a unified whole" (OUSD AT&L

August 2008). Components of a system by themselves cannot achieve the capability

necessary to complete the mission. By bringing together these related components,

systems can develop a unique, useable product or deliverable. In the military, systems

themselves are individual components to a larger system. This larger system is

considered an SoS by the acquisition community, which "is defined as a set or

arrangement of systems that results when independent and useful systems are integrated

into a larger system that delivers unique capabilities" (OUSD AT&L August 2008).   The

focus of this research is modeling an SoS architecture with independent but useful

systems that have their own organizational behavior, which will be discussed later in this

section. With the individual systems and an overall SoS program office, this situation

demonstrates that it is an acknowledged SoS.

There are four types of SoS according to the System Engineering Guide; they are

virtual, collaborative, acknowledged, and directed. First, a virtual SoS is established

when a central purpose has been agreed upon but there is no central authority present and

relies on the invisible mechanisms to sustain itself. In the second SoS type, a

collaborative SoS requires component systems to voluntarily agree and fulfill to a central

purpose, and key supporters provide some level of maintaining and enforcing standards.

Third, an "acknowledge SoS have recognized objectives, a designated manager, and

resources for the SoS; however, the constituent systems retain their independent

ownership, objectives, funding, and development and sustainment approaches" (OUSD

AT&L August 2008). Finally, a directed SoS is centrally managed with integrated

systems fulfilling a specific purpose. While an individual systems function

independently of the SoS, their operational mode of the individual organization is

controlled by the central authority and used directly to support the SoS. Of these different types of SoS, the acknowledged SoS fits well into most of the typical DoD programs currently being managed. With these concepts defined, the research can start to build upon this foundation to understand the current SoS environment and organizational behaviors through the use of simulation and model tools.

SoS Environment

For the acquisitions in the military, there are organizations such as program offices, testing agencies, operations, independent systems within an SoS, and product development processes (including requirement definition and funding) that must be considered throughout the acquisition cycle of SoS. As the warfighter and the enemy continuously update their methods, strategies, and technology, the Department of Defense (DoD) looks to their acquisition processes to better support their mission. According to DoD Instruction 5000.02, it depicts the normal DoD acquisition process (see Figure 3), which is standard procurement process for new concepts and developments. System engineers and acquisition managers are trained to use this lifecycle model when approaching an acquisition. But this traditional acquisition approach would not meet warfighter urgent needs in a timely manner or provide a thorough capability unless sufficient funding and resources were provided. Also, DoDI 5000.02 is focused on a single program or system type: not collections of systems.
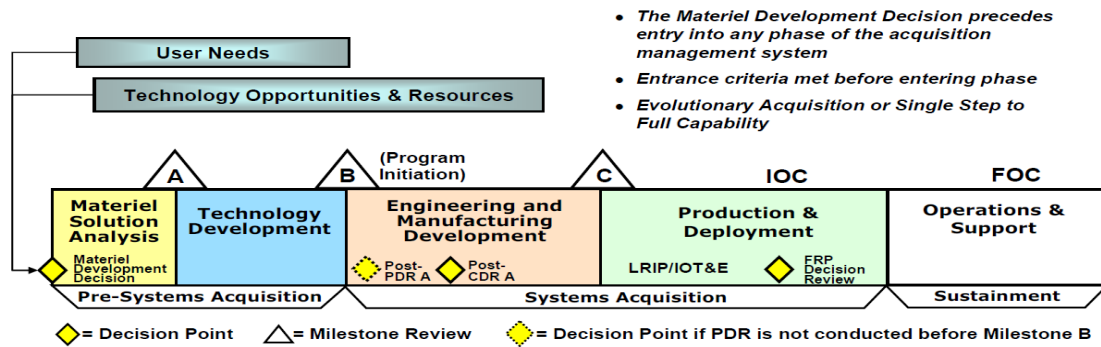
Figure 3: The Defense Acquisition Management System (DoD Instruction 5000.02 2008)

To meet the needs of the warfighter, the DoD prefers the evolutionary acquisition strategy to deploy capabilities rapidly to the user (see Figure 4). "An evolutionary approach delivers capability in increments, recognizing, up front, the need for future capability improvements" (DoD Instruction 5000.02 2008). To meet this challenge, system engineers must be able to break from the traditional point of view of establishing boundaries for highly stable and fixed requirements. But again, Figure 4 is focused on a single program or system. In reality, there are multiple, interacting programs or systems conducting the same evolution simultaneously. Being a system engineer on an SoS with multiple individual systems and tracking every requirements, specification, and design evolution for every piece in the SoS, it becomes incredibly hard for a system engineer to determine how small changes affect the overall performance of the SoS.
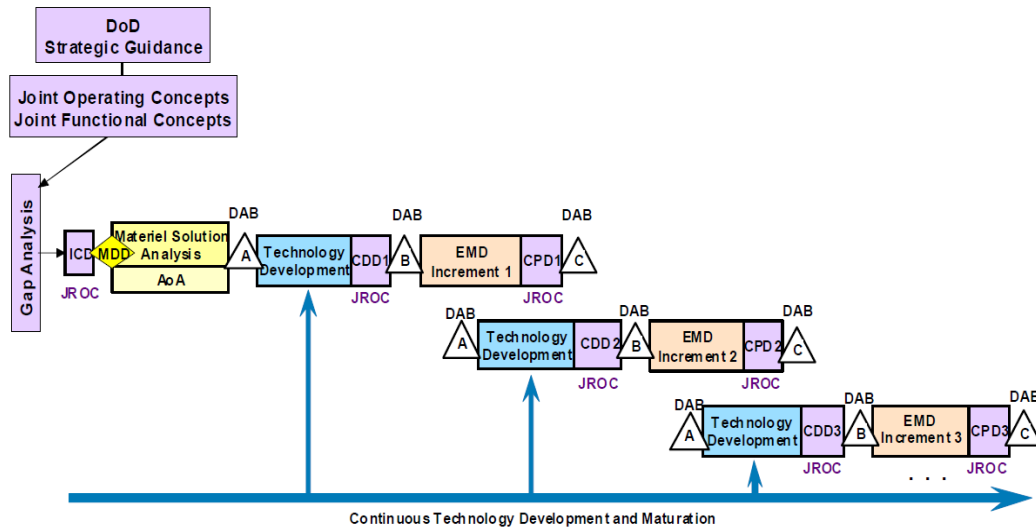
Figure 4: Requirements and Acquisition Process Flow (DoD Instruction 5000.02 2008)

"Today's defense SoS environment makes [the traditional] approach unworkable" (Dahmann, Rebovich, Lowery, Lane & Baldwin 2011). Dahmann, et al. describes how system engineers need to be prepared for external factors like funding cuts, priority shifts, and demands on SoS capabilities that can override the urgent needs from the user. These factors are known to happen during the acquisition process but not easily taught or prepared for. Additionally, Dahmann, et al. proposed that system engineers consider a process focused on core elements of system engineering that complement the evolutionary acquisition from DoD Instruction 5000.02 called the Wave Model (see Figure 5). Using principles from the Defense Acquisition Guide (DAG), system engineers can develop an evolving SoS architecture by implementing the following characteristics: multiple overlapping iterations of evolution, ongoing analysis, continuous input from external environment, architecture evolution, and forward movement with feedback. Finally, this particular "model applies particularly to 'acknowledged' SoS,"

11

which stated above is the focused SoS type of this research (Dahmann, et al. 2011). An acknowledged SoS is a type of SoS in which the overall SoS has its own processes, responsibilities and organizational structure while the independent systems that make up the SoS are responsible for their own systems and capabilities.
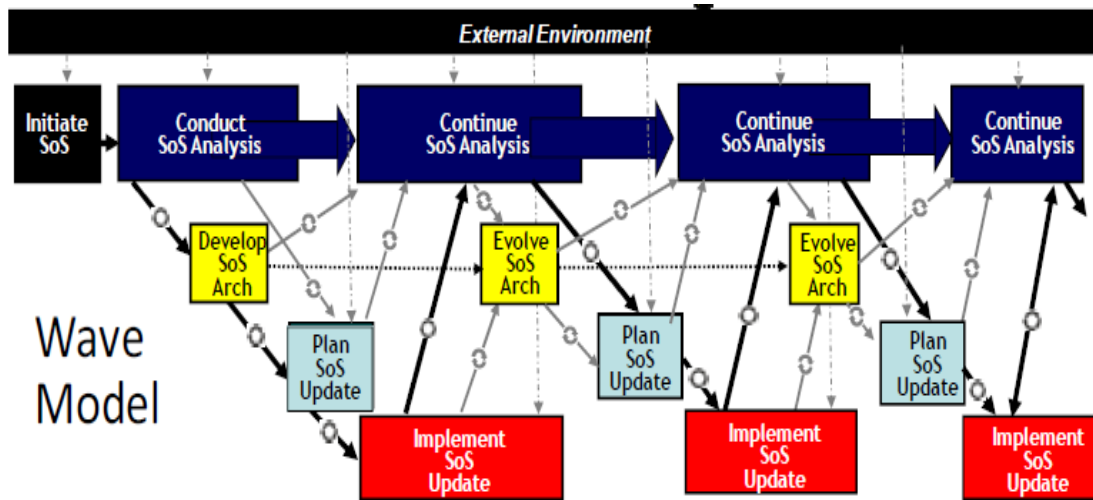
Figure 5: Wave Model (Dahmann, et al. 2011)

In recent years, the Remotely Piloted Aircraft (RPA) has undergone a rapid transformation from a reconnaissance aircraft with limited capabilities to a multi-purpose aircraft with high degree of integration with other military systems. As the RPA continued to prove successful and a source for future missions, "the traditional DoD acquisition processes and vehicles, in some cases, [were] expedited and/or [received a] waiver" (Delloiacono 2012). Delloiacono (2012) showed the evolution of initial concept of the RPA from a reconnaissance vehicle for commanders with limited equipment in the late 1990s to a killer scout in 2002 with munitions and increased communication equipment. From 2003 to 2011, the RPA underwent multiple upgrades to improve intelligence gathering, communication, and aerial performance. The RPA's evolution

from a technology demonstrator to an operational asset demonstrated the success of

expedited acquisitions but a failure from a system engineering point of view, because

sustainment and human factors were overlooked, which led to negative repercussions on

the SoS.   By re-evaluating the SoS continuously, the program office can limit these

issues but still provide the warfighter with the right capability.

Initiating Acquisition Process

To an SoS program office, their mission requirements are not static as

demonstrated above but are ever-changing to meet the needs of an unpredictable mission

environment. In the DoD, an SoS continually evaluates and updates its mission

requirements and capabilities to support the warfighter. The *Systems Engineering Guide*

for Systems of Systems provides the reasoning behind what initiates the change in an SoS.

Essentially, the "SoS systems engineer and manager review objectives and expectations

on a regular basis as the SoS evolves and changes occur in user needs, the technical and

threat environments, and other areas" (OUSD AT&L August 2008). To review

objectives and user needs, a systems engineer needs to receive inputs from other sources

such as feedback and external sources (i.e. stakeholders).   These sources include external

sources and feedback. "External sources that affect the SoS objectives, including the

stakeholder needs, the assessment of the threat, etc. Feedback on the feasibility in terms

of systems and their functionality, architecture limitations, and field experiences" (OUSD

AT&L August 2008). In the figure below, external influences, upgrades, and translation

of capability objectives trigger the SoS to assess current operational performance, which

leads to an evolution in the SoS architecture.
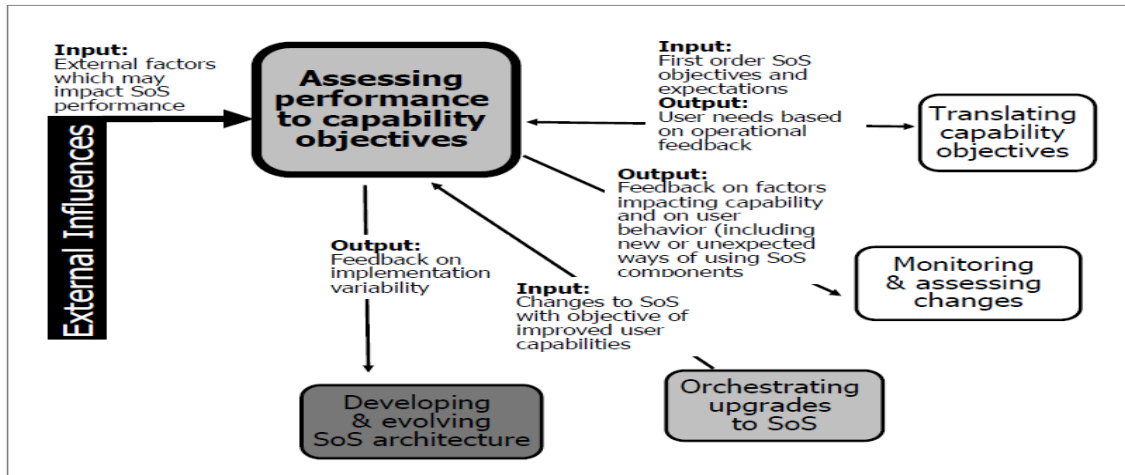
Figure 6: Relationship between Assessing Performance to Capability Objectives and Other SoS Core Elements (OUSD AT&L August 2008)

External and internal change requirements influence the composition of the SOS architecture. At times, the SoS in its current configuration will suffice to meet the needs of the warfighter, but the SOS program office, especially the SoS systems engineer, determines if current configuration needs to evolve. As a systems engineer analyzes the requirements, they determine how the individual systems should be connected and supported. With architectural and resource changes in the SoS, individual program offices associated with the SoS react to these factors based on their own organizational behavior.

Organizational Behavior Research

When examining a DoD acquisition, the fundamentals are presented and provide the core of what an acquisition program requires: requirements, funding, schedule, and performance. From an academic standpoint, these cover the basics, but there are intangibles that system engineers and acquisition officers need to understand when

handling real-world acquisitions. Most of these intangibles are linked to organizational behavior.

Hofstede's Cultural Dimensions

Organizational behavior can influence how an acquisition responds to a trigger. According to Geert Hofstede (1980), people are conditioned by a collective mental programming, which includes similar education and life experiences. This culture is shared by number of people, and together, they have built common forms of government, educational structures, and work organizations. Each country has developed their own unique national culture, which translates to how they view and interact within a given environment. According to Hofstede (1983), functions of distribution of power and control of uncertainty are provided by organizations. These can be provided through the use rules, policies, and standards. "It is not surprising, therefore that the functioning of organizations in a country and the way of thinking about organizations in that country are related to the country's position on the power distance and uncertainty avoidance scales" (Hofstede 1983). By examining a country's cultural values like power distance and uncertainty avoidance levels, organizational behavior can be established and modeled. Cultures, developed within a country, continue to be followed down to the organizational level. Hofstede's research into national cultures led to the development five cultural dimensions that account for many differences and similarities across societies, which affect the work environment.

The five cultural dimensions are power distance, uncertainty avoidance, individualism and collectivism, masculinity and femininity, and finally long- versus short-term orientation. "Power distance is a measure of the interpersonal power or

influence between [a superior and a subordinate] as perceived by the less powerful of the two" (Hofstede 2001). Large power distances demonstrate that subordinates are afraid to disagree with superiors and perceive an autocratic and less consultative decision making process. Small power distances exhibit a more democratic environment, where subordinates and superiors work together in a less hierarchical manner. Governing by majority vote "of decision making is rather unlikely to be practiced in complex work organizations: it would be feasible only if departments were autonomous and independent of other departments, whereas in fact modern work organizations are complex interdependent systems" (Hofstede 1983). The type of leadership that a manager displays could have an effect on the how well an organization performs based on their power distance index (PDI). If the leadership type, which will be discussed later in this paper, matches PDI of the organization, then this should have a positive impact on performance. "The boss-subordinate relationship is a basic human relationship," and the way that a system or business performs is based on this complex relationship (Hofstede 2001). For example, the manager demonstrates that they are open to suggestions from the project team, whose PDI is low and more consultative. The outcome will be a positive impact on the work environment, but unfortunately, if the manager is more authoritative, the impact will be negative.

Next cultural dimension is uncertainty avoidance, which determines the level of anxiety towards uncertainty or unclear requirements. Organizations "use technology rules, and rituals" in order to cope with uncertainty (Hofstede 2001). With these rules, organizations lower the uncertainty due the unpredictable conditions in their business and provide a stable environment. A high uncertainty avoidance index (UAI) tends to exhibit

higher stress, less risk tolerant, and follow the rules of the organization. The cultures with a strong uncertainty avoidance are "characterized by a higher level of anxiety and aggressiveness that creates among other things, a strong inner urge to work hard" (Hofstede 1980).   The lower that UAI is means that a national culture or organizational culture is less anxious when it comes uncertainty and more willing to take risks and willing to change. "Countries with weaker uncertainty avoidance tendencies demonstrate a lower sense of urgency," and known and unknown risks are acceptable like beginning new activities (Hofstede 2001). Cultures with higher UAI scores require more certainty before they are willing to take on the change. For an SoS, individual program offices with higher UAI scores should be less willing to change their current processes, which would lower the effectiveness of the SoS.

The third cultural dimension is individualism vs. collectivism. "Individualism implies a loosely knit social framework in which people are supposed to take care of themselves and of their immediate families only, while collectivism is characterized by a tight social framework in which people distinguish between in-groups and out-groups; they expect their in-group (relatives, clan, organizations) to look after them, and in exchange for that they feel they owe absolute loyalty to it" (Hofstede 1980). Depending on the level of individualism, it can affect how an organization and its members respond to rules and requirements, meaning that they could comply or not. Organizations with a higher collectivism score will be more dependent on others to meet the overall objective and willing to make decisions based on group consensus. "It is based not on self-interest, but on the individual's loyalty toward the [organization]—which is supposedly the best guarantee of that individual's ultimate interest" (Hofstede 1980). On the other side,

17

highly individualistic organization prefer individual decision making and are more

emotionally independent of the rest of group. The "relationship between the individual

and the organization is essentially calculative, being based on enlightened self-interest"

(Hofstede 1980). Hofstede stated that organizations, particularly United States

organizations, could get themselves into trouble due their inability to recognize the needs

of their employee. If organizations wish to remain at some level of individualism, the

organization will need to make the necessary adjustments to ensure the well being of their

workforce and maintain a certain level of performance.

Next, the fourth cultural dimension is masculinity and femininity, which

demonstrates how a society views goals, assertiveness, and what's important.

Masculinity and femininity "refer[s] to the dominant gender role patterns in the vast

majority of both traditional and modern societies" (Hofstede 2001). A higher masculinity

index (MAS) demonstrates a higher difference in values for each gender. These include

emphasis on achievement, advancement, individual decisions, and need for recognition.

Alternatively, countries with lower MAS considered life satisfaction and other job facets

like interpersonal relationships as more important. If an organization is more goal-driven

and in need of recognition, then they are more willing to seek out relationships in order

demonstrate their performance. This higher MAS score organization could conflict with

another low MAS score organization due to different their priorities. For purposes of this

research, it was important understand this cultural dimension and how it impacts a

society, but was not significant enough to be included in the simulation.

Finally, the last cultural dimension is long- versus short-term orientation, which is

a relatively new construct to Hofstede's cultural dimensions. This dimension

demonstrates what a culture deems more important, the present and past, or the future. The "long-/short-term orientation dimension appears to be based on items reminiscent of the teaching of Confucius, on both of its poles. It opposes long-term to short-term aspects of Confucian thinking: persistence and thrift to personal stability and respect for tradition" (Hofstede 2001). Long-term orientated societies tend to focus on perseverance, personal adaptability, and events to happen in the future. These dimensions also demonstrated a divide between eastern and western cultures, showing eastern countries with a higher long-term orientation and western and third world countries with a short-term orientation. This cultural dimension will impact organizations that interface with other organizations that vary on this dimension. Goals and strategic values are valued differently and will need to be carefully implemented to avoid conflict between organizations with a differing orientation. Overall, "organizations are symbolic entities; they function according to implicit models in the minds of their members, and these models are culturally determined" (Hofstede 2001). The most likely input for organization's behavior is linked to their national culture dimensions.

Gersick's Group Development

To understand group dynamics with diverse behaviors, Gersick (1988) demonstrated that normal group development (forming, storming, norming, and performing) to solve a project does not fit. Gersick's research illustrated that projects' timelines fall within certain phases. Her findings resembled a punctuated equilibrium, where there are long periods of inertia, followed by sudden change in stasis. No matter the timeline for the project, most groups followed a common lifecycle: first meeting, first phase, transition, second phase, and completion. The first meeting usually sets the

expectations and goals for the first phase of the lifecycle. "In phase 1, groups define most of the parameters of their situation quickly and examine them no further, concentrating their work and attention on only a few factors" (Gersick 1988). The common trend amongst the groups in the research appears around the halfway point in the group's lifecycle. As time becomes a limited resource, a new sense of urgency to complete the project ushers in a completion or abandonment of phase 1. A new direction and goals are confirmed and become the basis of phase 2. For groups, phase 2 becomes about solving the problem tasks. The transition point doesn't necessarily resolve intra-group tensions, but by this time the focus in task completion and not on collaborative decision making. Finally, the completion stage of the project focuses on the final meeting of the teams, where teams are editing or preparing their product for external use, focused on outside requirements, and more willing to express their feelings about the project and each other. This research highlighted the importance that leaders must plan their first meeting carefully. "Groups use the first meeting to diagnose the unique issues that will preoccupy them during phase 1" (Gersick 1988). As Gersick pointed out and Baldwin and Dahmann discovered, major DoD programs need to understand the relationships within the SoS, provide clear organizational coordination, and define the roles and responsibilities early, or the SoS will enter into a phase of punctuated equilibrium, which result in either team disbandment or refocus of goals with less time and more stress.

Leadership

Another important factor within an organization is the leadership. From an organizational point of view, leaders provide vision and set the direction. According to

Vecchio (2007), leadership and management are two distinct courses of action. "Management is about coping with complexity," while leadership copes with the changes to an organization or situation (Vecchio 2007). With an ever-changing environment, new requirements introduce a catalyst for an SoS change, but depending on the type of leader an organization has, this change can be handled with varying degrees of success. One of the studies that Vecchio applies describe that 65% of the variance in an organizational climate can be accounted by the style of the leader. "In his analysis of how complex organizations attempt to direct the behavior of their members, [Amitai] Etzioni identified three kinds of organizational power" (Vecchio 2007). These powers to obtain cooperation are coercive, utilitarian, and normative. Coercive is use of threats and/or punishment to gain cooperation with members. Utilitarian power is when organizations use incentives in order for members to conform to their directives. Lastly, normative power provides a sense of affiliation to its members in order to conform to a vision. Along with these powers, types of involvement were presented that members of organizations could possess: alienative (hostile, rejecting attitudes), calculative (rational, maximizes personal gain), and moral (committed due to social benefits that it will produce). In the table below, any "attempts to use types of power that are inappropriate for the type of involvement can reduce effectiveness" (Vecchio 2007). Based on the type of power employed by leader, the performance of the organization is dependent on if the leadership style matches the subordinates' involvement level. If the SoS or system leadership uses a normative type of power to influence subordinates but their type of involvement is considered to be calculative, then the overall performance of the SoS or system will be degraded. On the other hand a match between the types of power and

21

involvement like coercive and alienative would to be beneficial to the system or SoS when it comes to performance.

Table 1: Etzioni's Model of Power Involvement (Vecchio 2007)

Types of Power

	Coercive	Utilitarian	Normative
Alienative	X		
Calculative		X	
Moral			X

Types of Involvement

A final topic on leadership is moving from transactional leader to a transformational leader. Transformational leadership provides vision, a sense of mission, communicates high expectations, promotes rationality, and provides personal attention to employees. Transactional leadership promotes rewards in exchange for good performance, intervenes when standards are not being met, or avoids decision making entirely. "Organizations whose leaders are transactional are less effective than those whose leaders are transformational—particularly if much of the transactional leadership is passive management-by-exception" (Vecchio 2007). The type of leadership impacts how an organization performs, because "its presence [can be] felt throughout the organization and its activities" (Vecchio 2007). The goal for organizations in order to improve relationships within the organization and performance is to transfer from a transactional to a more transformational type of leadership.

Political Environment

Finally, the last construct related to organizational behavior to discuss is political influence. "It is not only possible but likely that the political process will not only drive such design factors as safety, security, producibility, quantity, and reliability, but even influence the choice of technologies to be employed" (Maier and Rechtin 2009). Depending on the amount of influence, this complex, influential factor causes the organizational behavior to respond how to meet their stakeholders' goals. Program managers need to understand this process and be prepared to cooperate along with this influence. "One of the surest ways to stop your own career is to reject a request from top-level management" (Vecchio 2007). The political influence from superiors, stakeholders, and the environment causes program managers to accept new requirement, regardless of available resources. For a program manager at the SoS and system level, coping with this influence is to figure out the needs of the stakeholders to best achieve success in an acquisition program office. By knowing what the stakeholders want (even if they don't know it), a manager can gain influence and greater support for their program and future endeavors.

Simulation Approach Evaluation

When researchers wish to test which factors that affect an organization's performance or a production line, changing conditions in the real world is not considered optimal. It would require time and resources that the organization cannot spare time and resources to test ideas, especially when it affects current performance baseline. "Modeling is a way of solving problems that occur in the real world" (Borshchev and

Filippov 2004). To be specific, simulation modeling allows organizations to map out

their process, identify potential problems, and simulate the current situation to gain

insight. Borshchev and Filippov (2004) describe four major simulation approaches to

model different systems and problems. First, "System Dynamics is the study of

information-feedback characteristics of industrial activity to show how organizational

structure, amplification (in policies), and time delays (in decision and actions) interact to

influence the success of the enterprise" (Borshchev and Filippov 2004). Based on the

amount of stock in question, a system dynamic model will take the policies of the system

and adjust the flow rate to meet the conditions of the environment. Housing occupancy,

where the economics, social issues, and laws provide input into a home owners' decision

to buy a house, is an example of a system dynamic model. Two aspects of system

dynamics to keep in mind are that the stocks are not individually indistinguishable and

"the modeler has to think in terms of global structural dependencies and has to provide

accurate quantitative data for them" (Borshchev and Filippov 2004). Second approach

was dynamic systems, which models the mathematical reasoning behind a system. "In

contrast with the [system dynamics], integrated variables here have direct 'physical'

meaning: location, velocity, acceleration, pressure, concentration, etc., they are inherently

continuous, and are not aggregates of any entities" (Borshchev and Filippov 2004). This

approach has a narrow purpose to solve a set of problems that can be entered into other

modeling programs.

Next, discrete event simulation was the third modeling approach. "Discrete-event

system simulation is the modeling of systems in which the state variable changes only at

a discrete set of points in time" (Banks, Carson, Nelson and Nicol 2010). This particular

modeling system better represents processes that have queues, resources, and service time. Borshchev and Filippov (2004) state that this modeling style describes the flow of entities and resource sharing of a system. Some of the different types of discrete modeling tools have market niches in the service industry, logistics, and manufacturing. "Discrete-event simulation has various world views (e.g., event-scheduling, process interaction, activity scanning, state machines, and other formalisms) that vary greatly in modeling flexibility and analytical power" (Chan, Son and Macal 2010). Using discrete event simulation to model the processes of a requirement change can lead the modeler to understand what affects the process, time it takes to meet the change, and the interactions between processes, but the entity like an organization and its behavior and interactions are harder to model in a discrete event simulation.

Lastly, the final approach discussed was agent based modeling. Agent based modeling "is a hybrid discrete-continuous simulation model with proactive, autonomous, and intelligent entities" (Chan, Son and Macal 2010). Also, Borshchev and Filippov (2004) differentiate between agent based modeling and other simulation models like system dynamics and discrete event simulation due to what level the behavior is defined: global or individual. System dynamics and discrete event simulation approaches establish a global set of rules that entities follow, where agent based modeling set the behavior rules at the individual entity level. By establishing the rules, the individual agents, which can range from tens to millions, interact with the environment and the other agents within the environment to provide a local behavior. The resulting local behaviors of the agents provide an emerging global behavior of the system. "The fundamental feature of an agent is the capability of the component to make independent decisions"

(Macal and North 2006). This ability to make independent decisions, based on the internal rules provided, at the agent level presents modelers an opportunity to replicate a complex, real world scenario more accurately. "The behavioral complexity of an agent is quite flexible and ranges from simple binary decisions (yes or no) to complicated human behavior or intelligence" (Chan, Son and Macal 2010). In order to properly model an acquisition system, an agent based model would be preferred, because the system is an acknowledged SoS. These individual program offices react independently to the environment and make own decisions based on their own organizational behavior. From list of the different simulation approaches, agent based modeling is better equipped to represent an SoS program office and its individual system program offices.

Summary

Since the DoD employs systems within acknowledged SoS, where the individual system program offices are independently responsible for their systems, an understanding of organizational behavior within and between organizations has to be better understood. With evolving SoS architectures based on different types of initiations, the organizations with their own cultures and leadership react to the SoS environment and other organizations differently to produce a level of behavior that impacts the performance of their system. System engineers evaluating an SoS must be aware and take into account organizational behaviors when determining the overall fitness of the SoS, not just the technical capabilities. Finally, agent based modeling simulation has been shown to respresent the individual agent characteristics and rules and have the flexibility to respond to environmental changes and interactions between different agents.

III. Methodology

Chapter Overview

The purpose of this chapter is present this research's process into how investigative questions will be answered. First, additional reasoning is provided for selecting agent based modeling as simulation type for this research over different types of modeling. Next, a description of a pedigreed process in which systems respond and evolve, given a set of rules and environmental conditions is described. This process gives the research a starting point from which to observe an SoS and its behavior. Once simulation type and process was determined, rules of behavior and description of how the systems will interact with each other and environment is explained.   After that, the process of the model is described in nine steps. Finally, a discussion on verification is presented.

Selection of Agent Based Modeling

In the previous chapter, an examination of the different types of simulation illustrated that agent based modeling would be the preferred method of modeling an SoS acquisition program. To establish a procedure to model an SoS program, an understanding and set-up of an agent based model needed to be explored. According to Middleton's article (2010), he states that system behavior is not dependent solely based on current variables but on the history of the system as well. Also, "system component interactions frequently take the form of adaptive emergent behavior, in which the system coevolves with the environment and other systems" (Middleton 2010). Middleton continues by stating that this implies a level of fitness with a selection of characteristics

to best fit within the environment.   Based on the current environment and historical

system data, a system (agent) within the model can react to current conditions and

determine its own level of fitness. These agents can be socially intelligent in its ability to

cooperate and/or coerce other agents. Instead of being omniscient of the entire system,

an individual agent knows only its historical data, agent interactions, and environment

input of the world. Given that organizational and emergent behavior can be complex,

simple rules can be provided for the agents in order to explain these complex

relationships. "This multi-disciplinary approach [agent based modeling] supports

representation of individuals with widely diverging belief systems and standards of

behavior, a virtual necessity in accommodating clash of cultures that characterizes the

human dimension in much of today's military operations" (Middleton 2010). In a

discrete event simulation, basic physical dimensions and capabilities can be modeled, but

the added dimensions of behaviors and interaction by the entities and processes would

require a series of complex equations and underlying logic. By establishing a simple set

of rules for an organizational behavior and their reaction to events and their interfaces, an

agent based model can be used to understand the complex interactions of an SoS.

Chan, Son and Macal (2010) also state agent based modeling provides an

opportunity to understand the nature of a system through simulation of an agent's

interactions with other agents and their environment. Agent based modeling allows

researchers "to simulate cascading effects rising from possibly minor local interactions,

experimentally examine tipping points, identify and explain beneficial or malicious

emergent behaviors, and more importantly design mechanisms to grow (and discourage

malicious) behaviors" (Chan, Son and Macal 2010). This type of simulation tool allows

the researcher to examine small effect on one particular system or a large effort on the

SoS. Through customizing the impact of the effect on the overall system, behaviors of

the system can be examined to determine their impact during an acquisition of a program.

This effect could be adjusting the performance of one system, a set of systems, or the

entire SoS. Also, Chan, Son and Macal (2010) present a few examples of agent based

modeling of how a simple set of interaction rules can demonstrate a complex, emergent

behavior of a system. First example was Conway's Game of Life model of signifying an

evolving system, where agents live or die based off the interaction rules concerning their

neighbors. This simulation presents an observer an opportunity to analyze a system

demonstrate complex behavior with a few interaction rules guiding it. Another

simulation example was the Reynolds' boid model, which simulates the movement of

leaderless groups like with birds and fish. Reynolds (1987) stated the motion and flight

of a flock of birds seems relatively simple but visually complex. To an observer, it would

appear that the flock's flight is under a highly centralized control. "Yet all evidence

indicates that flock motion must be merely the aggregate result of the actions of

individual animals, each acting solely on the basis of its own local perception of the

world" (Reynolds July 1987). To help explain this complicated flight motions, Reynolds

(1987) used three behavior characteristics, collusion avoidance, velocity matching, and

flock centering, with a set of rule of behaviors. Again, this complex movement and

multi-entity behavior was modeled through individual level interaction rules in order to

provide understanding to complex issue. With the focus of this research being the impact

of organizational behaviors with an acquisition of an SoS, the interactions of individual

program offices with other program offices are fundamental to understanding these

behaviors given by a change in the environment. With this in mind, this research will

continue with using agent based modeling as its primary method to study organizational

behavior.

Bak-Sneppen Model

   With the selection of agent based modeling as a simulation method, the next step

of this research was to determine how to set-up and implement a model that incorporate a

level of fitness, organizational behavior, and interfaces amongst the different individual

systems. The first model that meets the objectives of this research is the Bak-Sneppen

Biological Model. In 1993, Bak and Sneppen introduced a simple model that focused on

the biological evolution for a set of species, which eliminates the weakest species within

same ecosystem. The Bak-Sneppen Biological Model states that the "fitness of each

species is affected by other species to which it is coupled in the ecosystem" (Bak and

Sneppen 1993). Fitness or the fitness landscape was defined as "the ability of species to

survive as a function of their genetic code" (Bak and Sneppen 1993). Bak-Sneppen

Biological Model is analogous to system configuration changes that impact other

systems. This model consists of N number of species (systems), which are connected to

two other species. They are initially provided with a random fitness level that is

uniformly distributed from 0 to 1. At each time epoch the specimen with smallest fitness

is selected and chosen for mutation. This specimen is given a new fitness level based off

another uniform distribution from 0 to 1. Also, the specimen's two neighbors are

selected and provided a new fitness level like the specimen stated previously. With very

few rules, this model provided insight to the complexity of evolution within an ecosystem

and a fundamental model to expand upon into other areas of study. The author hypothesizes that system's evolve within the DoD/SoS environment.

Expanding on this methodology, Bartolozzi, Leinweber, and Thomas (2006) used the basis of Bak-Sneppen process, the evolving the lowest fit organism, and applying it to economics. They still selected the lowest fit specimen and provided it with a new level of fitness. Their modification to this process was the local interaction that a specimen or element has and receives from its two neighbors. By introducing an influence term from its neighbors, the element is now considered to have two fitness levels: local and global. In Bak-Sneppen, only the local fitness was considered the determining factor for evolution. In modified Bak-Sneppen, global fitness was the determining factor for selecting who evolves within the environment. This research showed that even the unfit specimens can produce a large enough global fitness to avoid selection due their interaction terms with more fit neighbors. "This inequality has a straightforward interpretation: species with a large number of connections will have a high barrier against environmental changes because they can rely on numerous resources" of others (Bartolozzi, Leinweber and Thomas 2006). This mutual cooperation increases a specimen, organization, or company chances of survival in ever-changing environment. The authors showed that by surviving on local fitness alone a specimen would need a fitness level of 74% to be considered fit for the environment. When using global fitness, the specimen requires only local fitness level of 48% to survive in an environment. With this in mind, fitness levels being calculated and observed in this research will be based on the global fitness for the system and the SoS within an acquisition environment.

Defining the Initiators

In Chapter II, an examination was conducted to determine the potential influences that an environment has on the acquisition process of an SoS. Two basic sources of change were determined: feedback and external sources. From the *System Engineering Guide for Systems of Systems*, feedback and external sources occur for reasons and achieve different capability objectives. With external sources, this source of change can be broken down into two areas (stakeholder needs and threat assessment), which focus on the overall strategic objective of the SoS. Through stakeholder needs, an SoS receives technological advancements and new or expanded requirements. From the author's experience, this particular external source is considered an upgrade initiator. This is a periodic environmental trigger that provides the SoS with enough resources and schedule to complete required activity and highly stabilized requirements. Since this is a known trigger that happens periodically to an SoS, the political and public influence, which will both be considered as contextual influence for the remainder of this research, will be considered low. An upgrade trigger can range from the next block of an SoS development to a new technological advance. The other type of external source characterized in this research is threat assessment. Based on the author's experience, this threat trigger is a time critical change to the system, set of systems within the SoS, or the entire SoS with high contextual influence and low resources. Since the time is a factor, schedule for performing the activity is less than the other triggers, and the requirements to implement the need are not as clear as the upgrade trigger due to unknown impact that the new requirement will have on the entire SoS. Finally, the last source of change is feedback, which can characterized as user need. The user need trigger comes from the

warfighter or a test unit that notices a deficiency of a system through operational use or testing. This deficiency is reported through their chain of command, which requests the change. The change request from the SoS's customers can potentially provide adequate resources, requirement, and time to the program office in order to complete the required activity. With multiple agencies involved, the contextual influence is considered higher than the upgrade trigger but not as high as the threat trigger due to urgency. Appendix A provides a definition for each of the triggers listed above. For the purposes of this research, these three triggers will provide the required stimulus within the model environment for the systems to respond based on their organizational behaviors. By inducing change within the SoS environment, the organizational behaviors of the individual systems will respond to create varying levels of fitness for each system.

Rules of Behavior of the Model

Within the model, the systems have established characteristics or attributes, which provide a basis for how they interact with other systems and the environment. Figure 7 presents an overview of the model interactions. When a trigger event occurs, the organizational behaviors receive these conditions and respond with a new local fitness. This local fitness is then supplied by their network neighbors' influence to contribute the individual system's global fitness within the SoS.
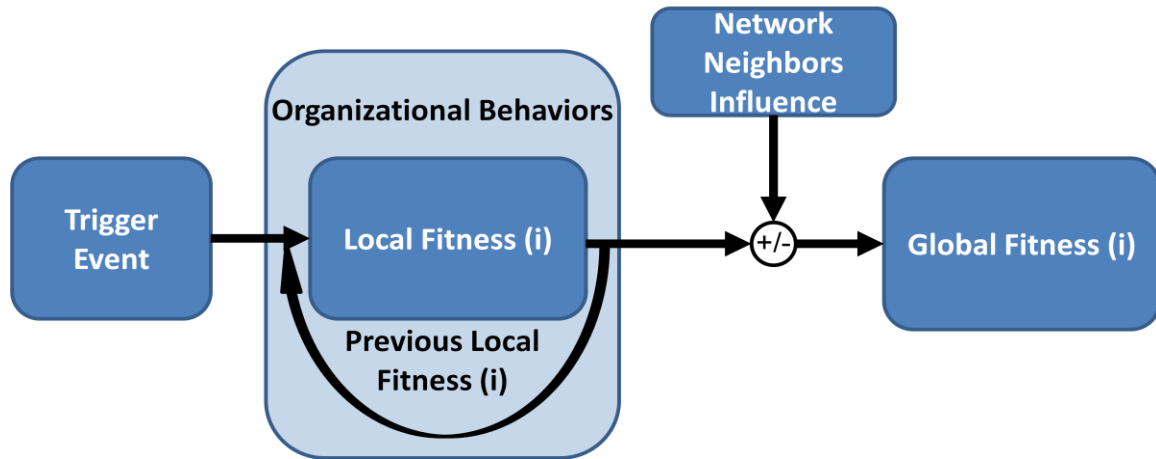
Figure 7: Overview of Rules of Behavior

Organizational Behavior

From this research's examination of organizational behavior, it was determined that cultural dimensions of PDI, UAI, and individualism/collectivism would best characterize the behavior of an acquisition organization. Since one of the assumptions of this research is that a DoD acquisition SoS being represented in the model, the last two dimensions of muscularity and long-term orientation would not have a significant impact since the systems would not vary from system to system with these characteristics. PDI, UAI, and individualism/collectivism are cultural dimensions that can vary significantly amongst the different systems. To determine these levels for each system within the model, each of these dimensions were triangularly distributed around Hofstede's determined level for the United States. Although the DoD culture could be considered different from the United States' culture, it was determined to be a reasonable starting point for the purpose of this research. For example, the United States' average PDI is 40, and by using the standard deviation, provided by Hofstede's data, the maximum and minimum values were determined by adding and subtracting the standard deviation to and from the given PDI. So, the triangular distribution for each system for its PDI was

set to a minimum of 18, mode of 40, and maximum of 62. This method was applied to the other two dimensions, which resulted in UAI and Individualism/Collectivism index of (22, 46, 70) and (66, 91, 116) respectively. With these dimensions determined, the rest of the system attributes can be discovered and derived. In Figures 8, 9, and 10, they graphically show the range of each cultural dimension for the research model.
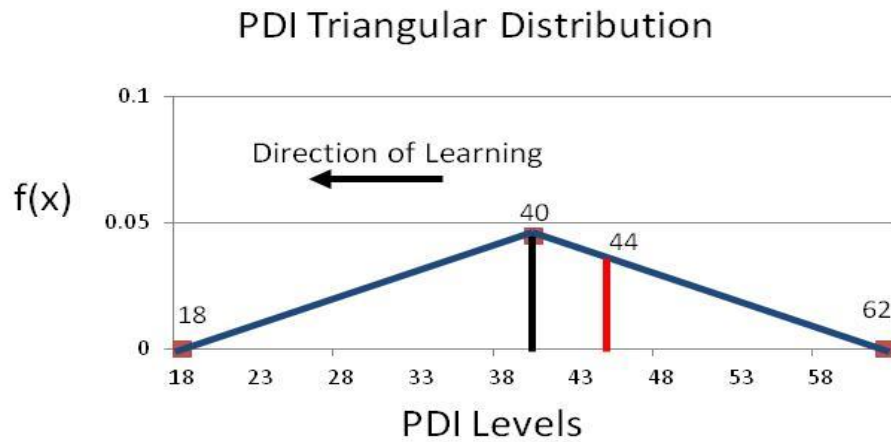

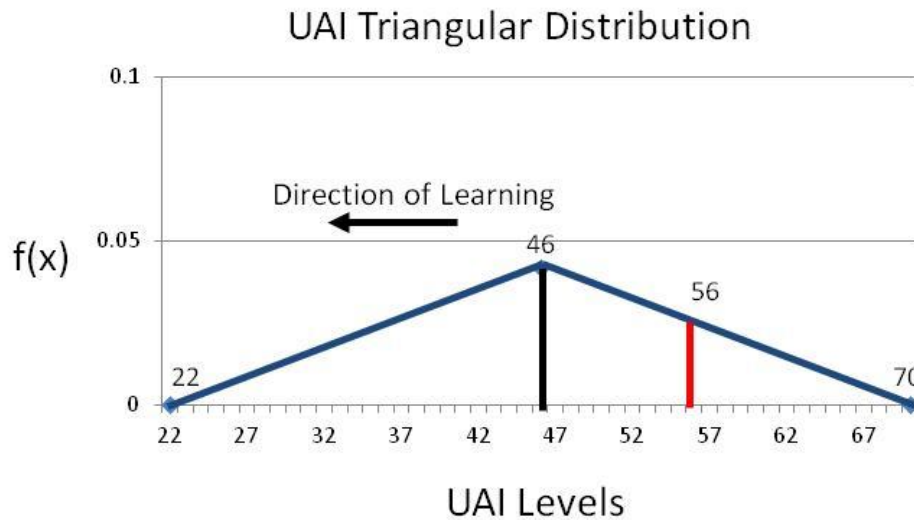**Figure 8: Power Distance Index Triangular Distribution**

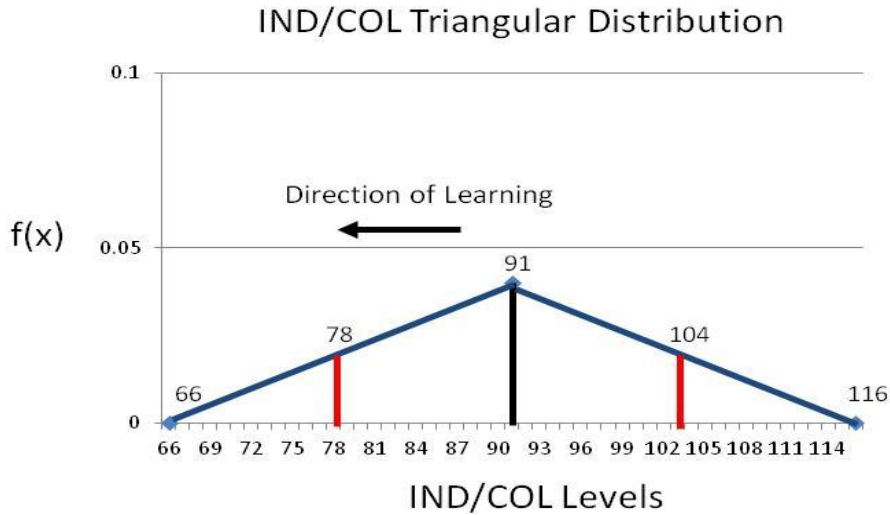**Figure 9: Uncertainty Avoidance Index Triangular Distribution**

Figure 10: Individualism/Collectivism Triangular Distribution

Another attribute for each system was the leadership factor, which shows the type of leadership that a system possesses. As examined earlier, the leadership and followership dynamic is an important relationship that helps determine the effectiveness of the organization. The leadership factor was divided into types of leadership: transformational or transactional. To determine this factor in the model, a uniform random number from 0 to 1 was determined for each system. If the random number was greater than or equal to .5, then the leadership factor was considered to transformational; otherwise, it was considered transactional. Next, ambiguity for the system was determined by taking one minus an index of requirements. This requirement index (0 to 1) is given by type of trigger presented to SoS environment. With cultural dimensions, leadership factor, and ambiguity established, three functions of leadership, ambiguity, and individualism provide the rules of behavior of the systems' attributes and environment trigger type.

36

Reaction by Organizational Behavior

The first function, leadership, determined if the leadership factor and the PDI value for each system were considered a match paired, thus this is an effective relationship within the organization. Figure 11, below, demonstrates the flow and interaction of the trigger characteristics and organizational behaviors into the conditional functions that develop the basis of willingness, selfishness and ability.
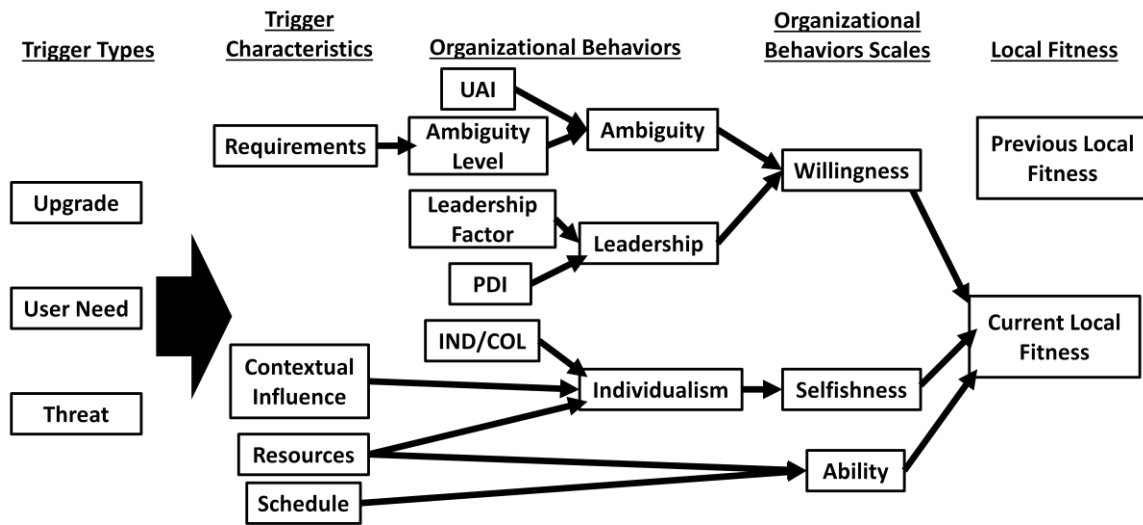
Figure 11: Overview of Interactions

A lower PDI describes an organization that is more open to debate and exchange of ideas between the followers and leadership. When an organization contains a transformational leadership and low PDI, this illustrates a matched pair and is effective from this particular point of view. This matched pair is similar to a high PDI and transactional leadership, where a high PDI organization follows the orders and recommendation of superiors without debate or additional input. In the cases where there is high PDI and transformational leadership or low PDI and transactional leadership, these cases are considered a mismatched and thus considered ineffective organizational leadership pairing. Since transformational leadership tries to help the conditions within

the organization for the followership, the leadership function for this case would result in

a slightly higher value than other mismatch. In determining the cutoff value for matching

PDI and leadership factor, Hofstede used a PDI value of 44. In Figure 12, Hofstede

plotted the PDI values versus individualism index, grouped the surveyed countries

according to similarities, and provided a cutoff line between the small and high PDI

countries. For the purpose of this model, a system with a PDI less than 44 and

transformational leadership style would have a leadership function equal to 1. This value

is also the same for transactional leadership style and a PDI value greater than and equal

to 44. Any mismatched pairing results in leadership function value of zero.
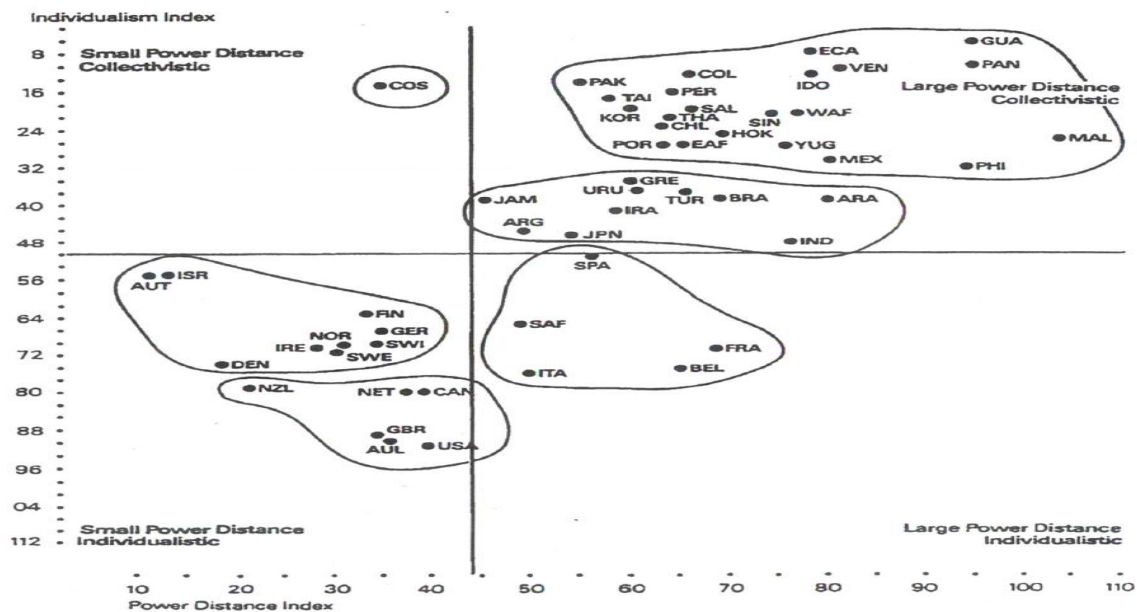
Figure 12: Power Distance/Individualism Index Scale (Hofstede 2001)

The second function for each system is the ambiguity function. Similar to the

first function, this function takes cutoff value for the UAI value, established by Hofstede,

and compares it ambiguity of the current environment. In Figure 13, Hofstede separates

countries with high and low UAI with a value of 56. Organizations with UAI values less

than 56 are deemed less anxious when reacting to an ambiguous situation, so the result of

38

this comparison function is at a minimum .25 for highly ambiguous situation and a maximum of 1. For higher UAI organizations, the maximum value is 1 again. This is due the environment providing more requirements to the program offices and lowering the ambiguity level. So even with a high UAI, systems, given sufficient requirements, will result in ambiguity function of 1. Once the requirements become less defined, these systems' ambiguity function will be significantly degraded. The maximum value that these systems could possibly achieve is .5, and the minimum value is 0, once the ambiguity is greater than .25. The biggest difference between the UAI and ambiguity level comparison is the smaller and larger UAI values. Smaller UAI values were given opportunity to provide a larger ambiguity function value due to them being less anxious in an ambiguous environment. Since they are less anxious, they can handle these unclear situations better than program offices with higher UAI.
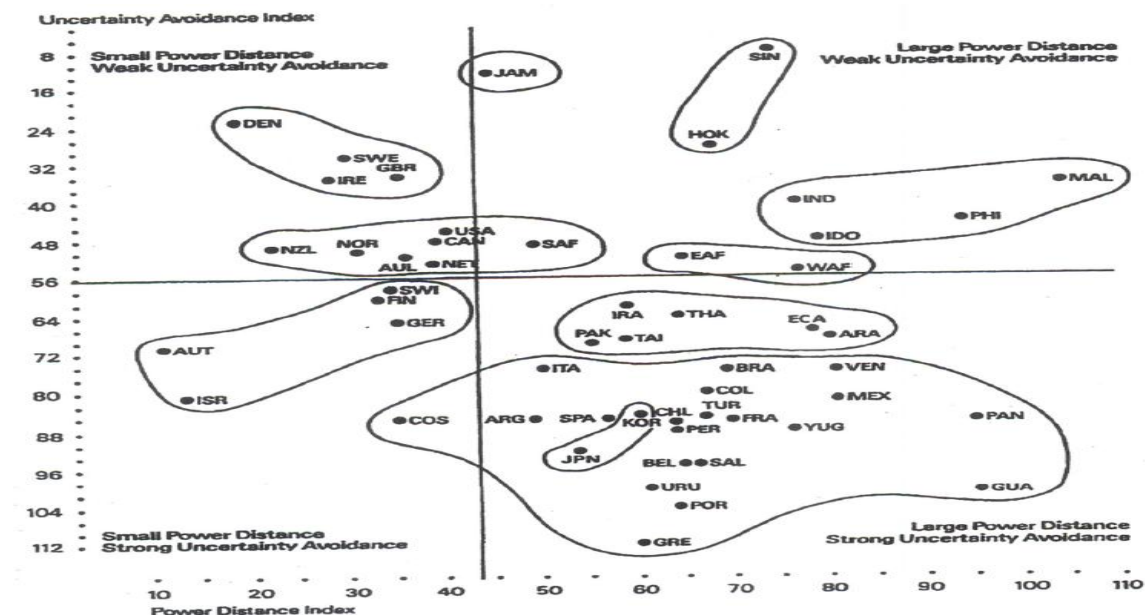

Figure 13: Uncertainty Avoidance Index/Power Distance Index Scale (Hofstede 2001)

The last function for a system attribute is based on system's individualism value, the contextual influence, and resources. Contextual influence and resources value are provided by trigger to the environment. The rule of behavior for the individualism function is based on how individualistic or collectivistic a system is. Low individualism values mean the system is more collective and is less likely to become selfish and share resources. The behavior of the organization is broken down into three sections of triangular distribution of individualism, based on the United States' score. It was determined that the more collective behavior ranges from values 66 to 78. The normal and highly individualistic ranges were determined to be from 78 to 104 and from 104 to 116 respectively. Since the United States' individualism score was the highest in the world, the triangular distribution did not cross Hofstede's critical value of 50. The 78 and 104 cutoff points were therefore determined by finding the halfway point between minimum and average values and average and maximum values respectively. From these three ranges and amount of contextual influence and resources, value of the individualism function can be determined, and in this case the lower value is better, which will be explained later in this section. When resources or contextual influence is high for the low individualistic range, the value would equal to lower value, but when the resources and contextual influence are low, this value increases slightly. In the medium range, higher contextual influence and resources result in a lower, but any low values from either contextual influence or resources or both will result in a higher value. For the case that both are low, the value will be higher than the other conditions within this range. Finally, the last range is considered highly individualistic and will contribute, which means a lower individualism function value, to SoS with only high levels of contextual influence

impacting the system's decision to support. Without a high level of contextual influence even with high resources, the highly individualistic system will not contribute to the SoS and result in a higher value. With function values and system attributes explained, these values provide the foundation from which SoS fitness and system fitness can be derived.

Willingness, Selfishness, and Ability

Following Middleton's approach (2010) with agent based modeling, a system's local fitness was determined through addition of the previous local fitness and the current from organizational behaviors. To obtain local fitness of each system, the organizational behaviors and environment characteristics need to be combined. "Measurement instruments that are collections of items combined into a composite score, and intended to reveal level of theoretical variables not readily observable by direct means, are often referred to as scales" (DeVellis 2003). These combined organizational behaviors and environment characteristics are now called organizational behavior scales. The organizational behavior scales are represented by willingness, selfishness, and ability of the system. According to *Webster's New World Dictionary, 2nd Edition*, willingness is the consent to perform an activity or state of readiness to support an activity. Next, selfishness is amount of being concerned with one's own interests and less about others. Lastly, ability is the power to do something either physically or mentally. Also, it means that one has the appropriate skill or talent to perform the required activity.

With these definitions in mind, the willingness component is composed of the leadership function, ambiguity function, and the contextual influence value. These values best represent the willingness component, because the leadership and ambiguity function ensure that the system was ready to meet the required activity. The leadership function

provides the level of approval to agree to change their system, while the ambiguity

function understands the organization's anxiety to the current environment. Next, the

level of selfishness equals to the individualism function. Since this function displays how

an organization will respond either collectively or individualistic to a trigger from the

environment, it was clear that this easily equates to a level of selfishness. This value is

subtracted from 1, so a higher individualistic organization will produce a higher selfish

level, and thus a lower contributor value to the organization behavior. For ability, this

component is determined through the addition of weighted values of resources and

schedule. These values are given by the trigger type and correlate to the definition,

provided above. Through resources (talent, funding, personnel, etc.) and schedule, an

organization can better support an activity if resources and schedule are available to the

organization, but if it does not have the proper resources or schedule, the organization

does not have the adequate ability to support.

Local and Global Fitness

Once the organizational behavior scales have been determined, the next step in

the process is to determine the local fitness. Similar to Bak-Sneppen, fitness for this

research is defined by how well an acquisition program office can adapt to the

environment on technical, fiscal (economic), and behavioral terms to meet the needs of

the warfighter and its stakeholders. To calculate local fitness, the values of

organizational behavior scales are weighted and then added with the previous local

fitness. This represents the fitness of the individual system. From the research, a system

or program office within an SoS is not isolated and depend solely on their own fitness,

but also the additional influence of its network neighbors. Next, the global fitness for a

system is determined by adding the system's local fitness and the system's two nearest

neighbors' local fitness, which are multiplied by amount of influence that have on the

system. Figure 14 visually demonstrates this relationship, which is similar to method of

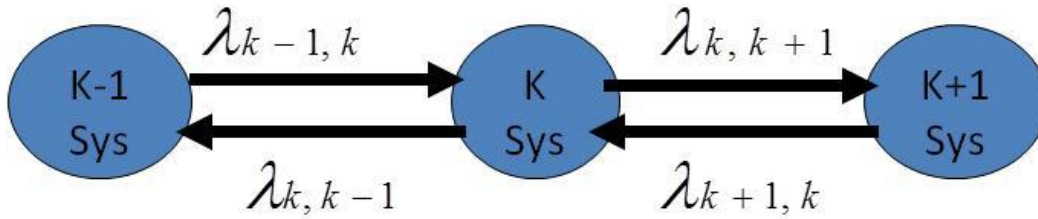the modified Bak-Sneppen stated earlier by Bartolozzi et al. (2006).

Figure 14: Global Fitness Calculations

Finally, the final output of the model is the average global fitness of the whole

SoS, which is determined by taking the summation of global fitness of all systems and

dividing by the total number of systems, at each iteration. Appendices B and D contains

a full list of the equations. By observing the average global fitness, it can be determined

how the SoS is performing within the environment, whether it is getting better, worst, or

staying the same over time. Here, the assumption includes that higher average global

fitness is better. In other words, all system are more fit to operate within the SoS

environment.

Process of the Model

Since the model will enable agent based modeling capabilities like environment

and other agent interactions, MATLAB was chosen to handle the numerous data

structures in the model, provide straightforward calculations when dealing with many

complex interactions, and support the experimentation process of this research.

"MATLAB is a modern programming language and problem solving environment" and

"a powerful tool for research and practical problem solving" (Higham and Higham 2005).
By building a variety of different arrays with the values of organizational behaviors and
trigger type characteristics described above, the proposed model can generate the fitness
levels for all systems and the global fitness for the SoS. Figure 15 presents the nine steps
of the overall model, which can be repeated for varying organizational behavior
influence.
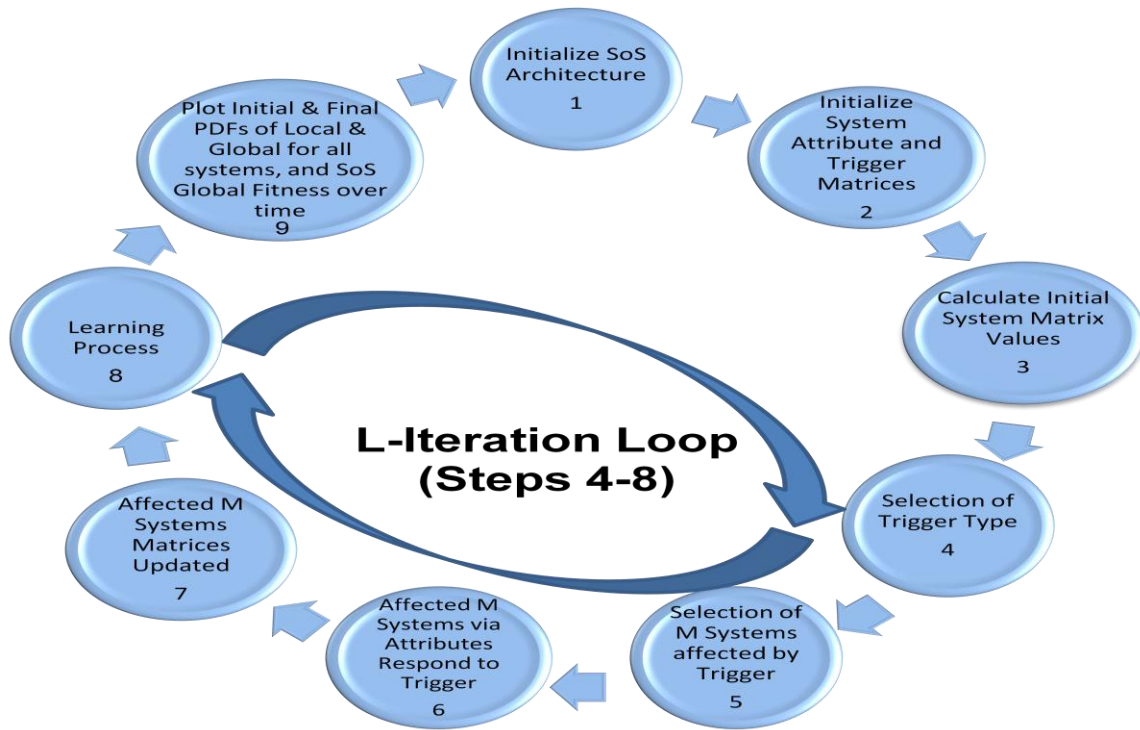
Figure 15: Simulation Process

On Step 1, the SoS architecture is initialized with the establishment of a fixed
number of N-systems, the number of L-iterations, and p-value for the SoS. The p-value
determines the amount of influence that the system's organizational behavior contributes
to the current iteration's local fitness level. Also, the matrices of each system, system
attributes, trigger set, and saved necessary outputs are established and initialized. Next,
the environment characteristics and system attribute the system attribute matrix are

44

initialized at L = 1 (prior to the first trigger) and provides the SoS environment a perfect start conditions with 100% resources, schedule, and requirements and 50% contextual influence. Although the environment may be considered perfect, the organizations with their behaviors may still end up with lower than expected fitness initially. The system attribute matrix is then filled out with the organizational behaviors and functions of leadership, ambiguity, and individualism in response to their own calculations (see Appendix B) and response to the initial conditions. Once the initial organizational behaviors and functions have been established, the initial system matrix, which includes willingness, selfishness, and ability, is next calculated to determine the initial local and global fitness for all systems, see Appendix D for calculations. The architecture of a 2 regular connection is established at this point. The global fitness of each system is determined through its local fitness and its two nearest neighbors' local fitness multiplied by their influence on the system. These influences were randomly determined in Step 1 to be a value from 0 to 1.

Step 4, the model begins the time iteration loop, where the process will continue return to this step after completed the necessary calculations for that particular numbered iteration until model has reached the established L^{th} iteration. For every iteration, the type of trigger will be selected and thus determined the values for the characteristics that the systems response to. The type of triggers was based on the acquisition initiators that were discussed earlier in this research. Based off the author's experience, an user need trigger, was determined to occur 50% of the time, while an upgrade and threat trigger was likely to occur 40% and 10% of the time respectively.   In Appendix C, it describes how values of the characteristics are calculated for the given type of trigger.

Step 5 selects a subset of N-systems that will be affected by the trigger. The selection is randomly conducted to about 33% of the systems in the SoS. The affected systems' organizational behavior then respond to new environment conditions on Step 6. After proceeding through the same equations and conditional statements from the initial state, these new values replace the initial values for that iteration. Once the new system attributes are established, Step 7 receives this new information and begins recalculate the affected rows and columns in the system matrix. New local fitness levels are then used to re-calculate the global fitness for all systems. Step 8 is the learning process, which takes known organizational behaviors and systems' local and global fitness and provides an opportunity for the weakest system to improve its organizational behavior. The model identifies the m-selected systems and determines the probability of learning. The ability to learn and improve their organizational behavior was based on the Rule of One-Eighth. In Pfeffer's and Veiga's research concerning the development of people within an organization to improve the success of the company, they stated one-half of organizations "believe the connection between how they manage their people and the profits they earn" (Pfeffer and Veiga 1999). Of these, one-half will attempt to improve their companies through human improvement measures. Finally, only one-half of those seeking improvements actually continue with their improvements long enough to detect a level of benefit. "Since one-half times one-half times one-half equals one-eighth, at best 12 percent of organizations will actually do what is required to build profits by putting people first" (Pfeffer and Veiga 1999). If the m-selected system meets the probability of 25% for improving, its PDI, UAI, and Individualism values are triangularly distributed again with same values as before. Each cultural dimension compared against its previous

46

value to check to verify if the value did in fact improve. If the value did not improve, then value was set to its previous value, so it didn't become worst. By having the process try to improve the organizational behaviors but fails, this meets the intent of the One-Eighth Rule. Next, the same or improved organizational behaviors are transferred to their columns in the system attribute matrix to be used for the next iteration. Steps 4 through 8 continue until the Lth iteration is complete. The final step of process is plotting of global SoS fitness at each iteration and the initial and final probability density functions (pdf) at local and global fitness for all systems.

Summary

With the selection of agent based modeling and pedigreed evolutionary method, this research was able to develop a set of rules of behavior and a process applicable to SoS acquisition. This methodology provided the researcher the ability to explore and answer the investigative questions, posed at the beginning of this research.

IV. Analysis and Results

Chapter Overview

This chapter provides answers to the investigative questions presented in Chapter I and analysis of the simulation and its results. Recall the three questions concerned with what affects an acquisition SoS, rules of behavior within an SoS, and how to calculate the local and global fitness of the systems. An examination of the simulation was conducted by presenting the evolution of the model, general observations of the final version, and focused analysis of a few systems.

Investigative Questions Answered

At the beginning of this research, three questions were presented to determine what affects an SoS, the rules of behavior, and the fitness calculation from willingness, ability, and architecture. First, the triggers that affect the SoS environment were concluded to come from two sources: external and feedback. Using a definition from *Systems Engineering Guide for Systems of Systems*, the external source was broken into two components of stakeholder needs and threat assessment. Based on the experiences of the author, these were characterized as an Upgrade and Threat trigger to the SoS environment respectively. Feedback was renamed as the User Need trigger. Their impact within the simulation model, which will be described in greater detail later in this chapter, demonstrated that the SoS environmental trigger and the systems' reaction to the trigger was the main source for the level of fitness by the system and SoS. When resources and schedule are more abundant during an Upgrade trigger, the organizational behaviors and their reactions to the environment provided a higher level of fitness for the

system and SoS. Additionally, lower resources and schedule during User Need and Threat triggers produced an overall lower fitness levels for SoS. Under a User Need trigger, the fitness levels depend more on the response from the organizational behavior. These levels are dependent on whether the organizational behavior is comfortable current situation or contain a better reaction, given the conditions. With this ever-changing environment, the model demonstrates that these triggers have an impact on the current fitness of the SoS. Although more resources and schedule would provide a better opportunity for a system to be more fit, it is not the sole source in its fitness determination.

The business rules for the organizational behaviors and the model, provided in Chapter III, were determined through the analysis of Geert Hofstede's research into the cultural dimensions of nations across the globe.   Of the five dimensions, three were determined to be of significant value: PDI, UAI, and Individualism/Collectivism. Each system was provided a certain level of the selected cultural dimensions through a triangular distribution around the current United States values. These cultural dimensions provided the organizational behavior for each system. Also, a leadership type for the system is determined through a random, uniform distribution, where half the systems are given a transactional leadership type, and the other half is provided a transformational leadership type. Between the PDI and leadership type, a leadership function can be determined through this relationship. For the other business rules of the model, the organizational behavior of each system interacts with the conditions of the environment like contextual influence, requirements, and resources. Based on their organizational behaviors, these interactions could have positive or negative effect on the system. These

relationships with other two organizational behaviors lead to the final calculations of the final functions, needed to determine the fitness levels of the systems and the SoS. Also, the simulation determines a selected number of systems to be affected by the triggers. These selected systems are provided conditions from the environment and later in the model given a chance to improve their organizational behaviors, similar to lessons learned of a program office. Due to the fact that some of the systems, not all, are provided additional resources by certain triggers, the non-selected systems' levels of fitness are slightly reduced, in essence they are considered not as important by the stakeholders to improve their systems for the customer within the SoS.

Finally, the determination of a system's fitness was presented at two levels: local and global. Local fitness was determined by the weighted combination of the system's current behaviors for the given environment and previous level of fitness for the each system. The organizational behavior scales consisted of willingness, selfishness, and ability. Willingness is composed of the functions leadership and ambiguity and contextual influence. By subtracting from one, the system's level of selfishness, which was composed of the individualism function, can be determined. Lastly, ability can be determined through the amount of resources and schedule that a system is given in order to complete the objective of the trigger. With these components and the previous system's fitness, the local fitness of the system can be calculated. By adding architecture to the formula, global fitness can be obtained. This research used a two regular connection architecture, where a system is only connected to its two nearest neighbors. To find the global fitness, an expanded version of the Bak-Sneppen method, discussed in Chapter III, was used. By adding together the local fitness of the system and a product of

its two neighbors' local fitness multiplied by their influence on the system, the global

fitness of an individual system was concluded. The SoS global fitness was determined by

taking the average of the all of the global fitnesses at each time iteration.

Evolution/Results of Simulation

The following section will discuss the evolution of the model during this research,

general observations of the SoS, and focused analysis on selected systems within the SoS.

The purpose is to understand the rationale for changing the model to its current state and

provide an analysis of the potential that organizational behaviors and environment have

on fitness of a system.

Evolution of the Model

Initially, the model consisted of the basic flow, described in Chapter III, which

included initializing and establishing required matrixes, providing system behaviors,

trigger selection and environment characteristics, reaction by the systems to current

environment, and finally the calculation of local and global fitness. This basic construct

of the model demonstrated that all of the systems were reacting to current environmental

conditions but was not converging to any particular fitness level. Upon further review,

the trigger selection was set to one type at a time, which revealed the average global

average of the SoS was only responding to the trigger type with no overall dynamic

behavior. In Figure 16, it displays the average global SoS fitness over time. It

demonstrates how the SoS reacts to the current environment conditions. When the

Upgrade Trigger is selected, the average moves upward dramatically, and the opposite

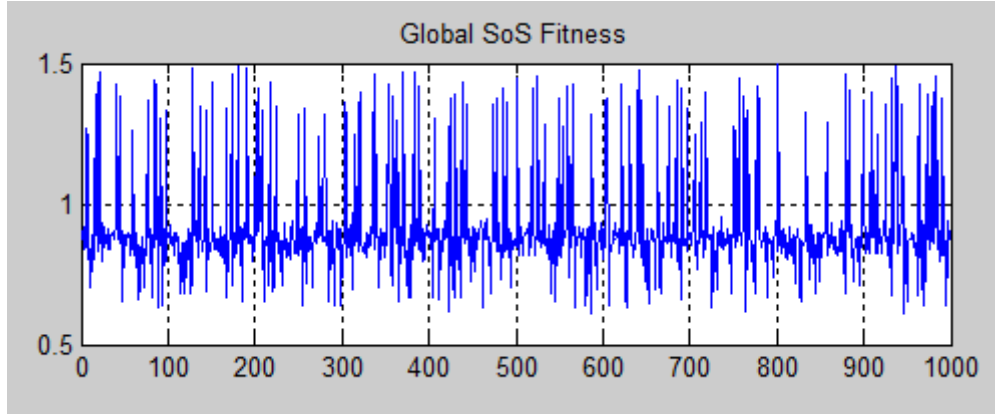effect occurs when the Threat Trigger is selected.

Figure 16: Average Global SoS Fitness of Basic Model

In order to make the model more analogous to an agent based model, previous fitness of the system was included in the local fitness calculation. Following Middleton's (2010) example, an agent based model uses its past history in combination of current conditions of the environment and its behaviors in order to respond to the present situation. By adding a weighted value of previous fitness of the system to the local fitness calculation, this met the intent of knowing the past, stated by Middleton (2010). Figure 17 displays a similar pattern as the basic model with peaks and valleys that correspond to the types of triggers affecting the environment. Overall, it appears that the memory of the last previous fitness helps keep the average within a certain fitness range for number of iterations before the consecutive extreme trigger conditions caused abrupt changes in the average fitness levels.
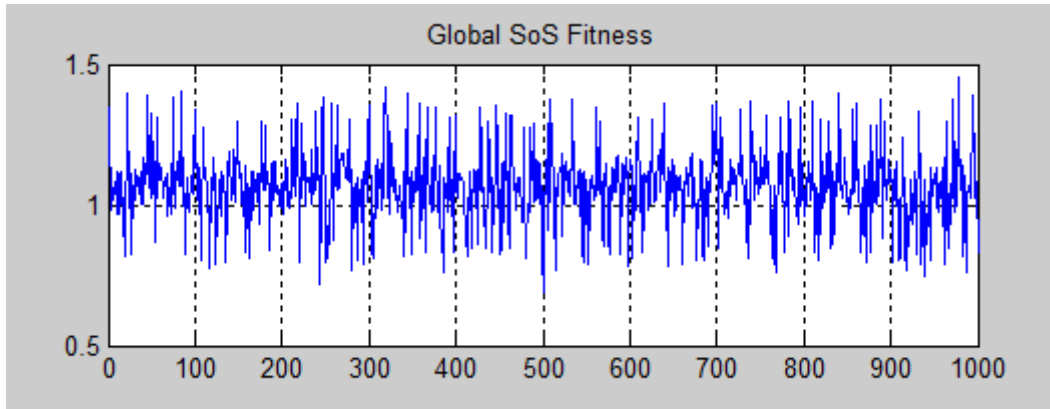
Figure 17: Average Global SoS Fitness of Previous Fitness Addition Model

The next evolutionary step for this model was selecting a random subset of the SoS rather than selecting the entire SoS for a requirements change. From the author's experience within DoD acquisition, not all of the systems within an SoS are told to upgrade the performance of the system due a change request from the warfighter and/or stakeholder. Additional code was written into the model to represent this selection process. Currently, a third of the possible number of systems is only selected randomly for a possible change in their system. Although only a selected number of systems are reacting to the current environment, the overall fitness profile was similar to the original model and provided no additional dynamic behavior. Also in this step, a varying weight for the organizational behavior scales was added to the local fitness calculation to investigate the sensitivity of organizational behavior. The purpose was to determine impact that different levels of organizational behavior has on the SoS. This impact is discussed later in this chapter. Figure 18 displays the average global fitness of the SoS at a lower fitness level on average than the previous versions of the model. With the p-value for the figure is .6, it shows the organizational behavior still reacting to

environmental conditions and previous fitness level maintaining some level of control, so

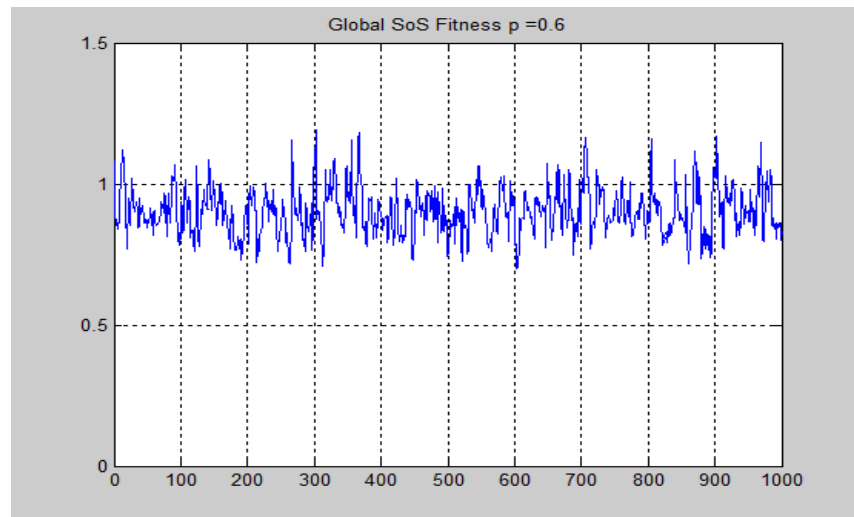extreme conditions do not overly affect the global fitness of the systems.

Figure 18: Average Global SoS Fitness with P-Value and Random Selection Addition

For the next version of the model, the establishment of perfect acquisition

conditions for the initial environment and a learning process for the systems were

introduced. Initially, systems were not provided with initial environment from which

they could respond to but the random uniform numbers for the missing reaction values.

To provide a more realistic setup, requirements, resources, and schedule were set at

100%, similar to the start of a DoD acquisition program. Additionally, the systems

within the model were given the opportunity to improve their organizational behavior.

Early versions of this learning process selected the system with lowest global fitness.

This quickly evolved into selecting the already affected systems and a 25% chance of

learning, based on the one-eighth rule as discussed earlier. This version of model

reduced the variability for fitness and began to trend upward and remains above one

throughout the majority of the iterations, see Figure 19 for the average global SoS fitness.

The learning process provided for the systems to improve their behaviors has caused the

overall average to trend upwards, but the Threat Trigger and its conditions to the environment still has an impact at times.
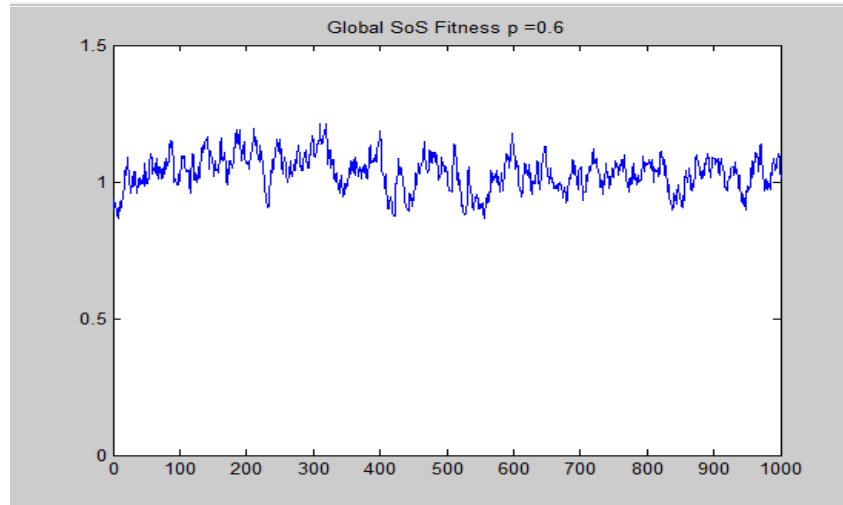

Figure 19: Average Global SoS Fitness with Perfect Conditions and Learning Process Addition

For systems not selected by the trigger, there is a potential for degradation of the system's local fitness, because they are not improving their performance or gaining potential resources and influence with the stakeholders. This degradation is based on a conditional relationship of the type of trigger introduced to the environment. For User Need and Upgrade Triggers, this results in a degradation of the local fitness for the non-selected system by 2.5% and 5% respectively. With these two types of triggers, the benefits would outweigh the potential negative impacts to the system fitness, so by not being selected, it causes slight degradation to the system for not improving and obtaining additional resources. For the Threat Trigger, the benefits would not outweigh the negative impacts, so their local fitness will remain the same as before. In Figure 20, it shows that once the first series of iterations are completed it repeats the pattern in the previous version. Slowly, the variation between the peaks and valleys become smaller and stays mainly above one while still reacting to the environmental conditions.
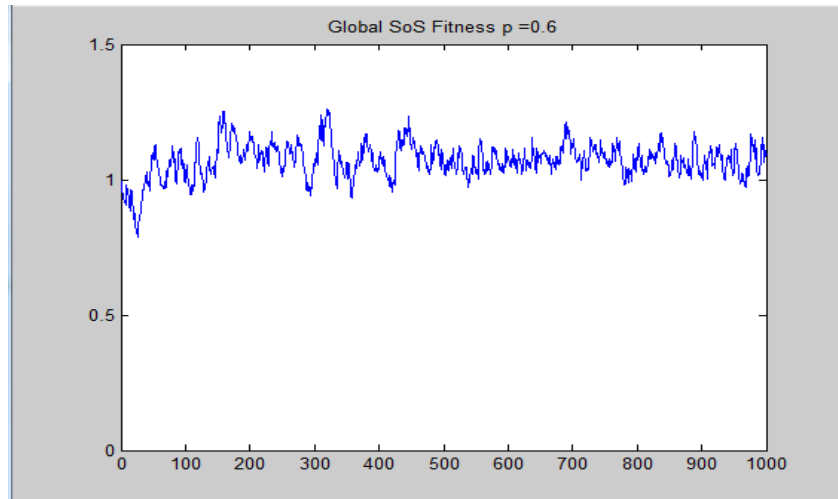
Figure 20: Average Global SoS Fitness with Degradation Addition

The last adjustment to the model was changing the ability levels of the non-selected systems. Upon reviewing the data, it was noticed that some ability levels of non-selected systems were not changing over long period of time. Since the ability accounts for 30% of the organizational behavior scale for the local fitness calculation, this would mean that systems that were selected earlier by a Threat Trigger would continue to hold a low ability level until it was selected again and may increase. If not selected over time, the system would continue to have a lower local fitness even though they could have good organizational behaviors. It is a local assessment by the program office of how it thinks it will be able to support the next iteration. This re-evaluation selects the non-selected system's ability and adjusts the ability ±5%. The process could provide a system with an increase in ability or a decrease. Figure 21 displays the average global fitness for the SoS with this adjustment. With a few more downturn areas than the previous version but consistent capacity to return to its stabilized average global fitness level of about 1.1, this demonstrates a model with greater variance with the combination of environmental

conditions, degradation of non-selected systems, change of ability levels, and learning process.
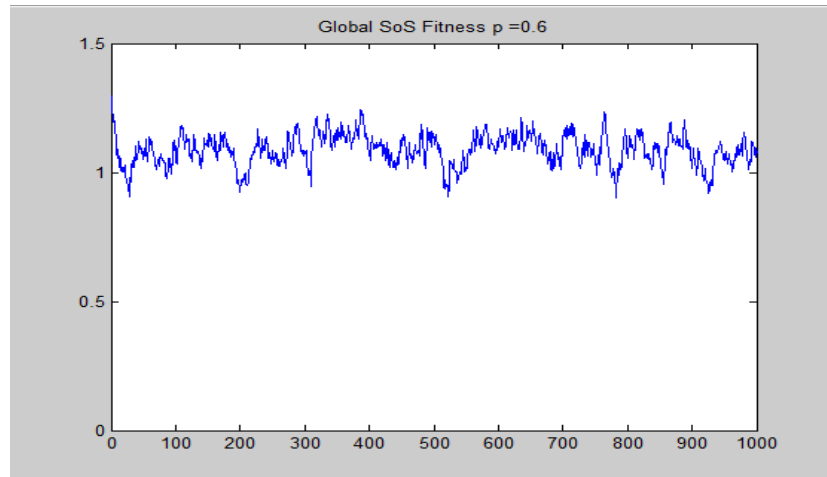

Figure 21: Average Global SoS Fitness with Ability Assessment Addition

General Observations

After finalizing the current version of the model, an analysis was performed on the data points and graphs that the model produced. The purpose of this section is to provide observations and determination of what is happening during the course of the simulation. First, the pdfs of the local and global fitness that were provided by MATLAB were examined to determine evolutionary changes from the initial condition to the final condition and for the different values of p. The p-value scale was introduced in the third version of the model, where different values of p acted as weighted influence for the organizational behaviors and the complementary p-value (1-p) was used as weighted influence for previous fitness. In Figure 22, it displays the initial fitness levels at the local and global level. These graphs are the same shaped pdfs for all p-value initial conditions. The top graph shows a fairly uniform distribution of local fitness. For the global fitness, a bell shaped graph is illustrated.
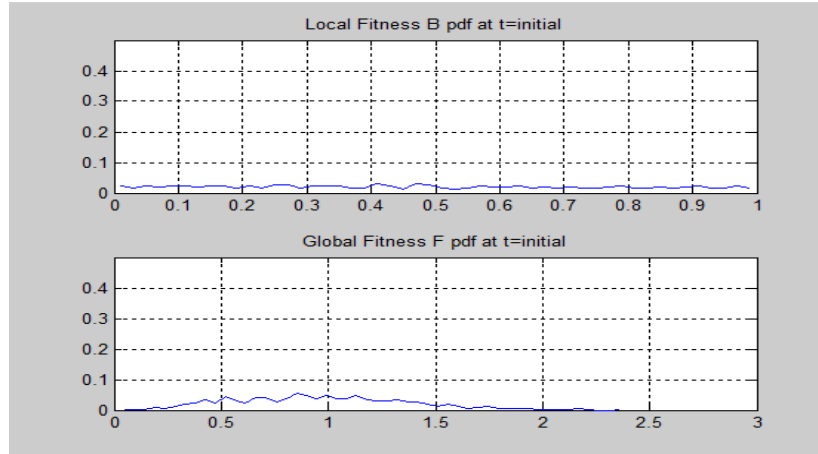
Figure 22: Initial PDF of Local and Global Fitness

By comparing the initial and final pdf graphs, it can be determined if there were any changes to the systems' local and global fitness over time. Figure 23 shows that the pdf of the local fitness for the systems is no longer shaped as uniform distribution but appears to have more of a normal distribution shape. The global fitness also differs by reducing the end points and increasing the density around one, so the variance of the systems' global fitness has decreased by the end of the simulation.
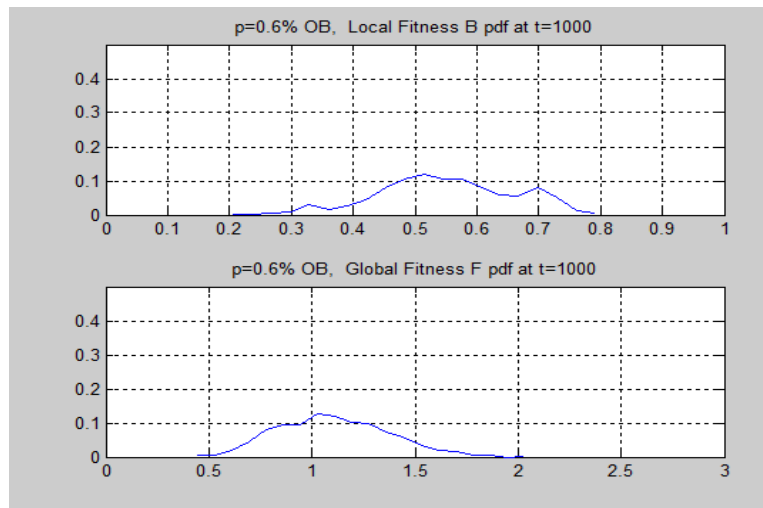

Figure 23: Final Local and Global Fitness (p=.6)

The figure above provides an example of local and global fitness pdfs at one particular value of p, .6. The p-value in the model is the amount of influence that

organizational behavior contributes to the local fitness. With higher p-values, the more

influence that organizational behavior contributes to local fitness over the previous

iteration's local fitness, and more influence previous fitness has when p-value is low.

From the pdfs, that was provided by the simulation, the different p-values showed similar

distribution like in Figure 23 but their densities shifted to higher fitness levels as the p-

values increase. The exception in this case was when p = 0, which implies the local

fitness depends entirely on the previous fitness levels. This resulted in the local and

global fitness approaching zero due to the degradation of the systems and no

organizational behavior interaction over time. By introducing a p-value of .2 to the

model, the distributions are no longer approaches zero and produce two "bell shaped"

curves, which Figure 24 displays. The difference between .2 and .6 p-values is the mode

of the local fitness distribution slides positive. The global fitness levels show the similar

distribution pattern but the majority of the fitness levels were much lower in .2 value than
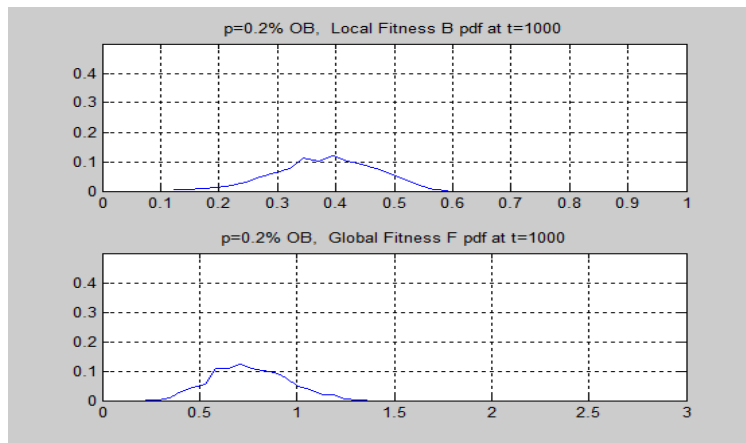
the .6 value.

Figure 24: Final Local and Global Fitness (p=.2)

In the figures below, they provide a variety of shapes in the local fitness pdfs of the other p-values. For the most part, the global fitness pdfs shift to right as the p-value increases. The local fitness pdfs appear to be shifting as well as the p-value increases, but when p=1, the local fitness does not have a bell or triangular shape as the others. Without the previous fitness level to contribute to the current local fitness, the organizational behaviors cause the system fitness levels to fall into three areas of the pdf (low, medium and high), which are related to environmental triggers. From these pdfs, a balance of influence between previous fitness and organizational behaviors towards the local fitness can be determined. When too little organizational behavior influences local fitness, the system cannot respond to the environment and provide an assessment to observers of the current condition within the system. With too much organizational behavior influence, the memory of past fitness levels is forgotten, and the systems' local fitness is determined solely by their reaction to environmental conditions, which in Figure 27 shows a formation of densities within three areas.
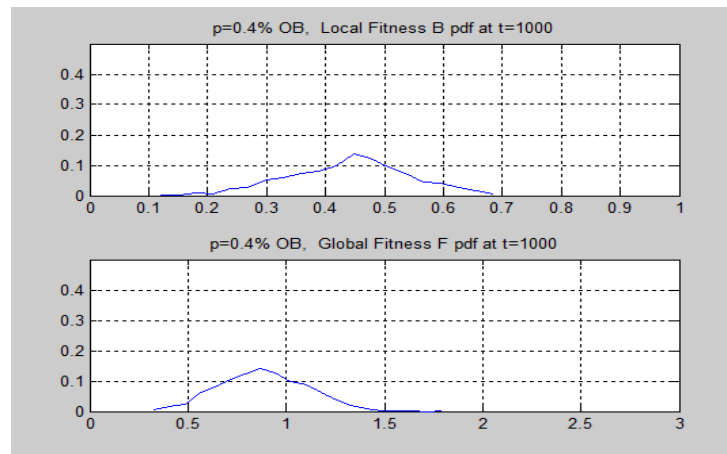

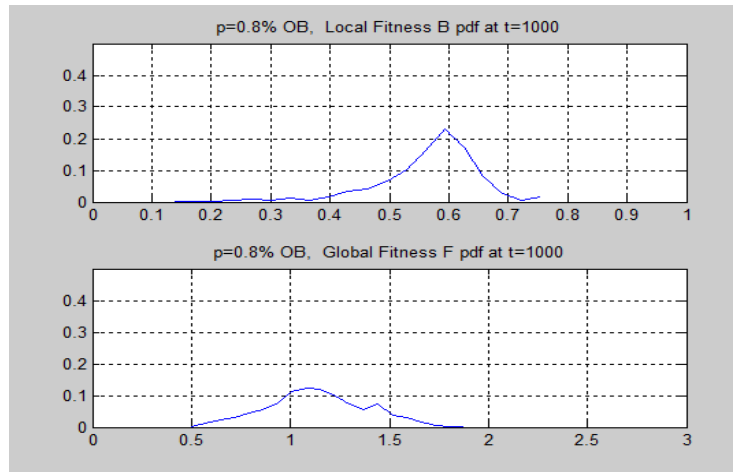**Figure 25: Final Local and Global Fitness (p=.4)**

Figure 26: Final Local and Global Fitness (p=.8)
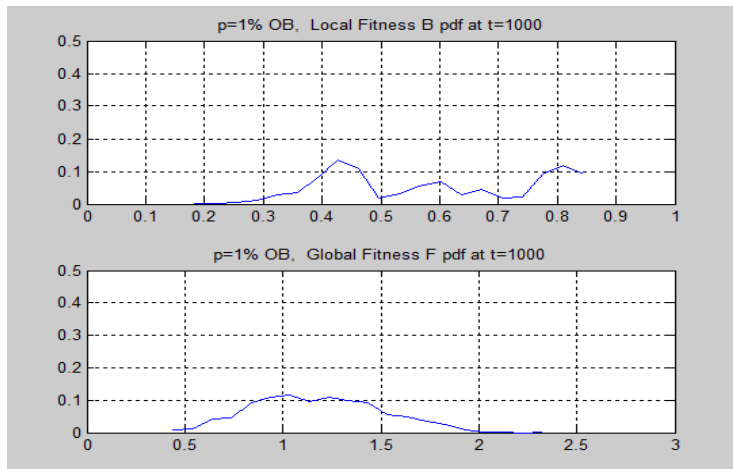
Figure 27: Final Local and Global Fitness (p=1.0)

Another interesting depiction was the average global SoS fitness over time. In
Figures 28 and 29, they display the average global fitness of the systems within the SoS
for a given p-value for that simulation run. Like before, the 0 p-value over time ends
converging towards a zero global fitness level. At the .2 mark, the average stays around
the .75 level. This shows how much organizational behavior influences the fitness by
increasing from zero to around .75 with very small increase in the p-value. At p-values
of .4 and beyond, the global fitness lingers around a fitness value of 1 to 1.2. As the
previous fitness influence decreases, the variation within the averages increases. After

61

examining the graphs from p-value of .6, .8, and 1.0, it was noticed that the variation

between points increases. At the .6 level, the line between points appears to be more

controlled, while the other two p-value graphs contain more of higher degree of

fluctuation over time. This is due to the lack of influence of previous fitness has in

determining the fitness levels for each system.
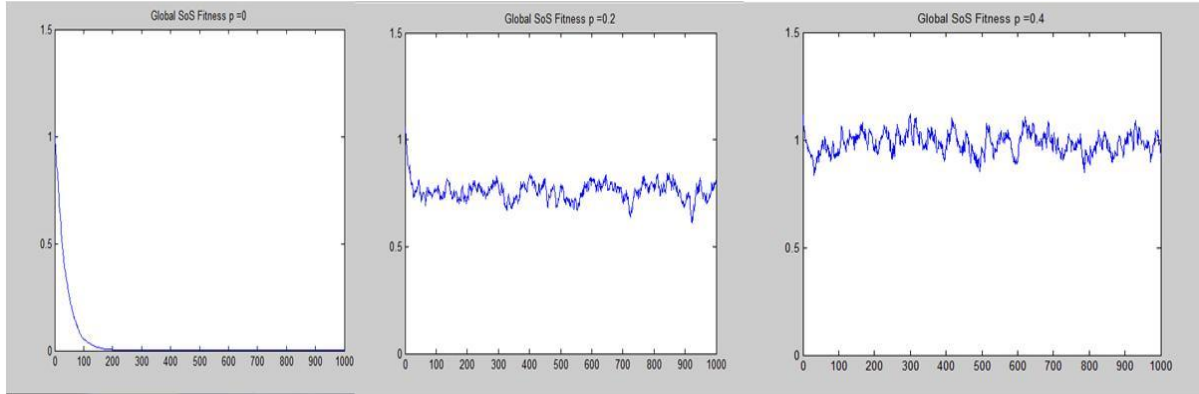
Figure 28: Average Global SoS Fitness (p=0, .2, .4)
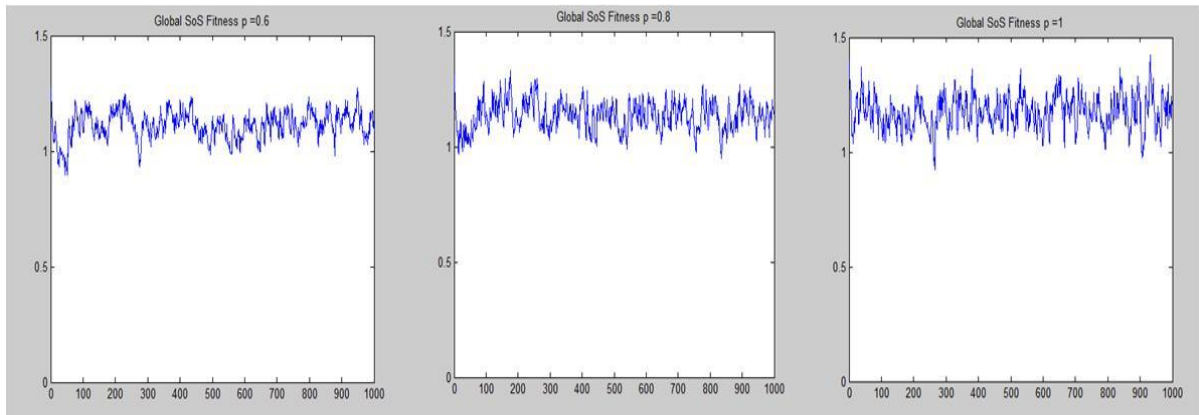
Figure 29: Average Global SoS Fitness (p=.6, .8, 1.0)

Another area of interest was the variance of global fitness amongst the system

over time. In the beginning, it was expected that the variance would be large, but it was

not clear if the variance amongst the systems over time would increase or decrease.

Figure 30 shows that variance amongst the systems was lower overall as time progressed.

After the initialization phase, the variance dropped below .08. There were times within

time cycle that the variance returned to initialization levels, whether it due to environmental conditions or degradation on number of systems. They still trended downward after that occurrence. By improving behaviors over time, variation within the SoS for the most part lower than the initialization phase of the model.
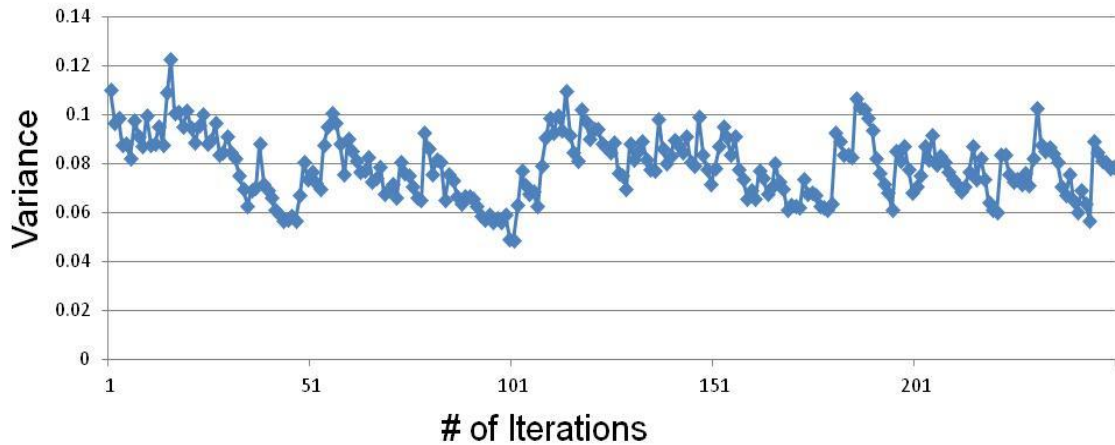
Figure 30: Variance over Iterations

The next point of interest is the moving average of global fitness, which was calculated by adding together all of the averages up to the current time iteration and dividing it by the current iteration number.   In Figure 31, the different p-values start at high fitness and drop dramatically to a certain fitness level. For p-values .4 and above, they reach this low point and proceeds to increase and converge to new stabilized fitness level.   Also, the gaps between the different values are smaller, and the range amongst these p-values is about .25. After the initial drop, the moving average for the p-values displays a similar pattern and by 125th iteration they reached their convergence fitness level. From this graph, it shows the perfect conditions that were initialized at the beginning of the simulation quickly drops away. This means that the combination of environmental conditions, degradation of non-selected systems, and poor organizational behaviors are decreasing fitness of the systems. For most of the p-values and after a

63

period of time, the systems through improving their organizations and some good

environmental conditions were able to increase their global fitness. Overall, the moving

average is converging to a global fitness level and achieving a stabilized fitness level

after period of time within the SoS and the environment.
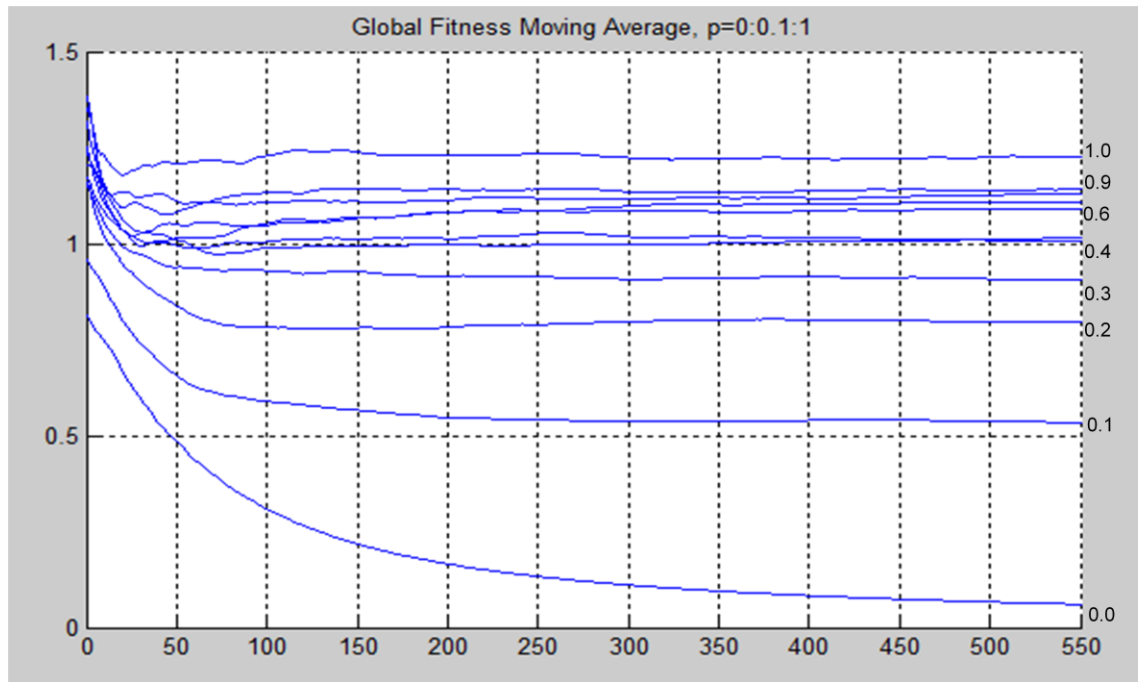
Figure 31: Moving Average for p-value from 0 to 1

To determine if the organizational behaviors and scales are significant to the local

and global fitness, each of the behaviors and scales were tested for correlation with both

local and global fitness. Based behavior values and scales on 50 systems over 250

iterations, Table 2 shows the correlation values for each value and scale. The correlated

value must be close to one or negative one in order for values to be considered correlated

to each other. The majority of values are near zero, so their influence towards local and

global fitness is not considered significant. Two values of willingness (W) and Ability

(A) correlated with local fitness, highlighted below, are above .5 and are the only ones

that are considered significant.

64

Table 2: Correlation of Behavior Values and Fitness

N = 50, L=250	
Correlation PDI and LF	-0.11606
Correlation PDI and GF	0.010436
Correlation UAI and LF	-0.06253
Correlation UAI and GF	-0.11522
Correlation IND and LF	-0.05501
Correlation IND and GF	0.007856
Correlation W and LF	0.547691
Correlation W and GF	0.23109
Correlation 1-S and LF	0.106276
Correlation 1-S and GF	0.060086
Correlation A and LF	0.534585
Correlation A and GF	0.222302
Correlation PDI and W	-0.16494
Correlation UAI and W	-0.18775
Correlation IND and 1-S	-0.00555
Correlation TT and LF	-0.01552
Correlation TT and GF	-0.01263

With these correlated values in mind, a regression model was developed using these values as the independent variables and local fitness as the response. First step was putting willingness, ability, and product of these two in a fit model against local fitness. On the surface, the regression model seemed acceptable with willingness and ability being significant and a reasonable R-square. Unfortunately, the normality assumption of the residuals was not validated, and the residuals formed a bow shape (see Figure 32) instead of a straight line when plotted on a normal probability plot. The skewness and kurtosis of this distribution was determined to be at -0.872 and 0.949 respectively, which means the distribution shape is not considered normal and the weights of the tails influence the overall distribution. This indicates that transformation of the response is required, so different transformations were attempted like performing a square root, logarithm, square, and reciprocal on the response but the normality was not validated.
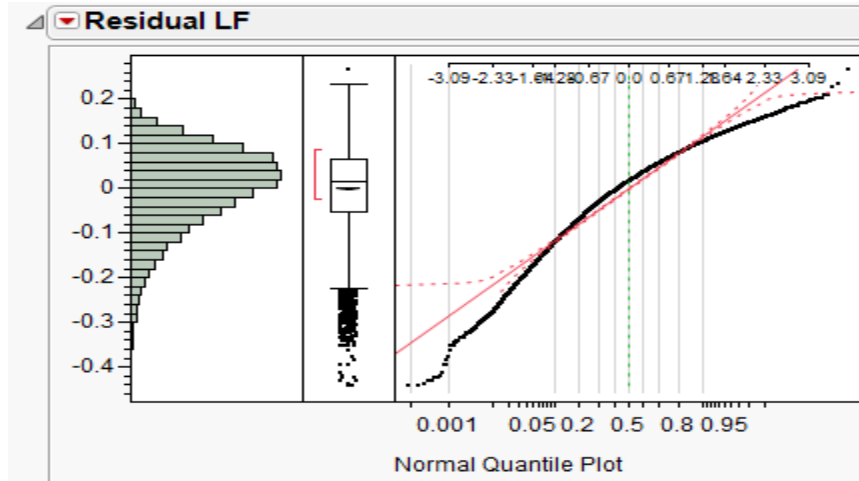
Figure 32: Normal Probability Plot of Residuals

Finally, using the Box-Cox Transformation in the JMP program, it provided the

best possible transformation to reduce the error within the model. By applying this

transformation, the lack of fit and total error of the model decreased, but the cross term of

willingness and ability was found to be significant. Upon removing that term, there was

no major difference between the two models with total error and R square terms were

equivalent to each other.   Additional validation of the normality assumption for the

regression model was applied. Figure 33 and 34 display the normal probability and

residual by predicted plots for reduced-transformed model. The normal probability plot

shows a fairly straight line. Skewness and kurtosis of the distribution decreased from the

original model to -0.559 and 0.324 respectively. This signifies that the residuals form a

near normal distribution, which is acceptable for the validation of the normality

assumption of residuals.
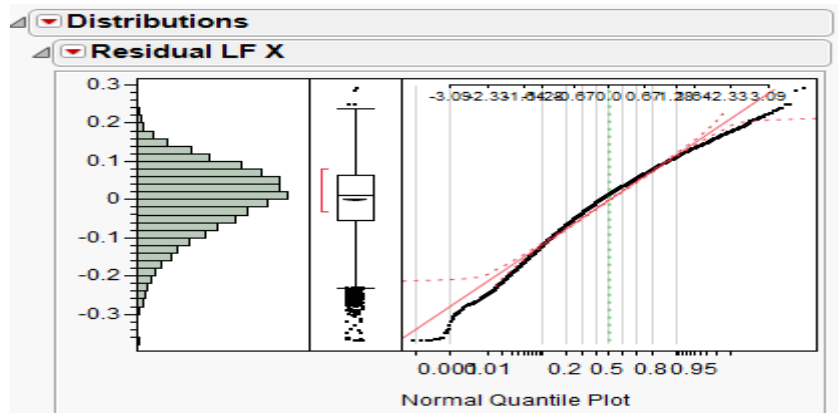
Figure 33: Normal Probability Plot of Residuals after Box-Cox

Figure 34 presents the summary of results for the reduced-transformed with analysis of variance and parameter estimates for the regression model. Overall, the model is acceptable, there is a large amount of error unaccounted for, but model is significant, which means probability of being greater than F critical value is less than 0.0001. The reason for the large amount of error is the fact two factors are not accounted by the model: previous fitness and selfishness. Selfishness was shown in the table above to not have high level correlation with local fitness, so it was not added to the model. Previous fitness was found to have a high level correlation, but this is reasonable since previous fitness and the current fitness are fairly close in value and thus are closely related to each other. Both willingness and ability are considered significant within the model.

Summary of Fit

RSquare	0.362271
RSquare Adj	0.362169
Root Mean Square Error	0.090863
Mean of Response	-0.56215
Observations (or Sum Wgts)	12500

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	2	58.61065	29.3053	3549.554
Error	12497	103.17595	0.0083	Prob > F
C. Total	12499	161.78660		<.0001*

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	11471	97.49237	0.008499	1.5342
Pure Error	1026	5.68358	0.005540	Prob > F
Total Error	12497	103.17595		<.0001*
				Max RSq
				0.9649

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	-0.874562	0.005256	-166.4	<.0001*
W	0.2695946	0.007588	35.53	<.0001*
A	0.1328766	0.003973	33.44	<.0001*

Figure 34: Summary of Regression Model

With this regression model equation, the local fitness can be better predicted by using

organizational behavior scales of willingness and ability. So, the final regression model

equation is:

$$LF^{1.6} = -.87462 + .265946*W + .1328766*A \tag{1}$$

where:
LF = Local Fitness
W = Willingness
A = Ability

Focus on Selected Systems

In the final analysis of the simulation, the systems with largest, average, and

smallest global fitness value were selected. The point of this analysis is to determine the

impact of intersystem weights have on determining where a system ranks amongst other

systems. Figure 35 illustrates that the three systems at the local fitness level are relatively

close to each other. From this figure, there doesn't appear to be any real significant

separation between the systems. When the same systems have their global fitness

68

plotting over the same number of iterations, the three systems display a significant separation. In Figure 36, system 45 has the highest global fitness initially, followed by system 38 and 20, respectively. Throughout the simulation run, system 45 maintains a clear distinction to the other systems as the largest fit system for almost the entire time. The other two stay within their appropriate ranges of medium and low throughout the run. An investigation to understand this logic proved that the weights of their neighbors were the source of this arrangement.
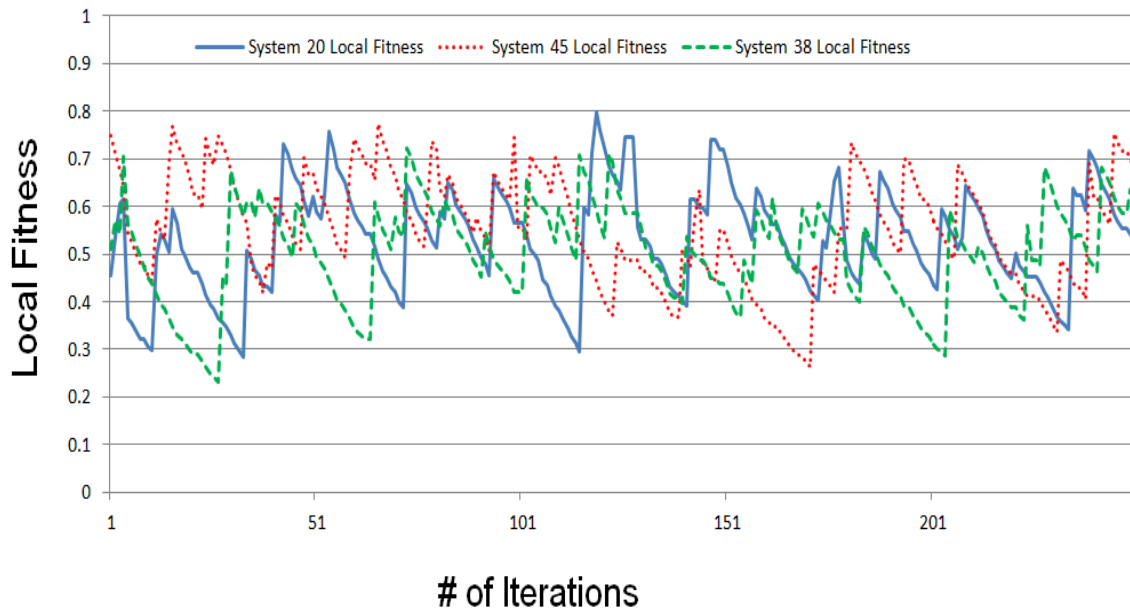
Figure 35: Selected Systems' Local Fitness
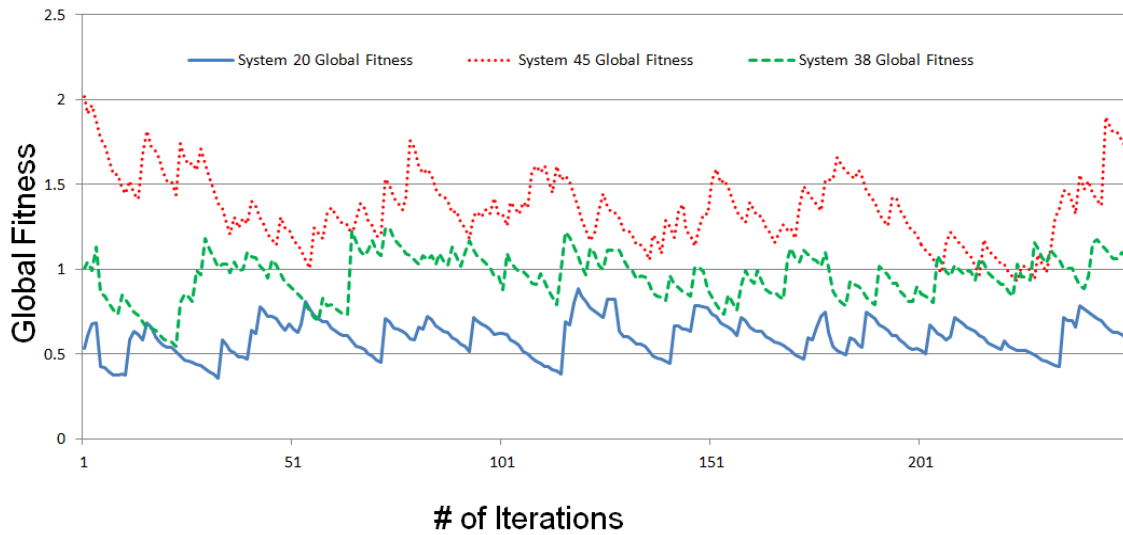
Figure 36: Selected Systems' Global Fitness

For system 45, its neighbors influenced it by contributing 0.6987 and 0.9792 of their local fitness to its global fitness. System 38 received 0.1438 and 0.7826 of its neighbors' local fitness towards its global fitness, while system 20 received lower percentages of its neighbors' local fitness. Based off the amount of the influence provided by their neighbors, these systems' global fitness will either remain at high, low, or medium levels. System 45 remained at a high level, because its neighbors will amount to contribute more to system 45's global. This is the reason why system 20 stayed low throughout the simulation run. Also, this is the reason why system 38 remains at medium fitness levels and becomes more fit than system 45 at times. For the purpose of this research, understanding the influence of the connecting systems has on a system's fitness is important for a program manager and system engineer to know.

70

Summary

Over the course of this chapter, the investigative questions were answered that described what initializes a change of requirement, rules of behavior within an SoS, and the calculation of local and global fitness. Next, a discussion of how the simulation evolved throughout the research and reasoning for the changes. Finally, analysis of the data was conducted with observations of finding the balance between previous fitness and organizational behavior. From the moving average, a punctuated equilibrium was discovered for all p-value levels. Willingness and Ability were determined to be significant through correlation and linear regression analysis. Lastly, it was determined that the weights placed on the connecting neighbors was significant contributor to determining of a system's global fitness level.

V. Conclusions and Recommendations

Conclusions of Research

In determining how fit a program office system is during the acquisition process, two organizational behavior scales and the amount of influence from connected program offices were discovered to be the critical factors. The two scales were willingness and ability, which are consisted of lower organization behavior values like leadership factor, PDI, and UAI. Ability was built upon the amount of resources and schedule that the system received to complete the change requirement. These two scales were shown to have correlation with local fitness at 0.548 and 0.535 respectively. From these two scales, the local fitness of the system can be more accurately predicted by program managers and system engineers. The model also demonstrated that improving organizational behaviors within a system leads to higher fitness for the SoS. These improved behaviors of PDI, UAI, and Individualism contributed to each system responding to the environment as best that it could. Obviously under good (or great) conditions for the program office, the behaviors responded very positive to these conditions, but under poor conditions and after some improvement, they provided the best response that they could provide in order to meet the requirements. Lastly, the amount of influence from other systems keeps the systems globally fit within the SoS. From the examination of specific systems, an average system was able to become more fit than the fittest system within the SoS at times mainly due to its connection to a greater influencing system. Within a SoS environment, it is possible for a system or program office to have a low local fitness to the point that it would most likely be eliminated from

the SoS or defunded by the government, but it becomes a stronger program office due to higher contributing influences from its connected neighbors.

Significance of Research

For the DoD acquisition community, the ability scale is already considered an important part of the acquisition process. In the research's model, the ability scale contained only resources and schedule, which are known acquisition parameters. Unfortunately, the willingness scale is not considered an integral part of the acquisition process. By understanding this behavior scale, it provides critical insight into how an organization responds to the environmental triggers. Acquisition leaders have a responsibility to improve their organization's behavior and provide an environment to meet the requirements of the stakeholders and the warfighter even with poor conditions within the environment. Acquisition leaders are given courses on leadership, acquisition process, and risk mitigation factors, but the understanding of how organizational behaviors influence an organization's ability to meet the requirement within an SoS is not fully explained. UAI is not about a risk mitigation strategy, but the level of anxiety that the organization has towards uncertainty. No matter what the mitigation strategy is, an organization will still have a high anxiety towards a request with very little clear requirements, and it will impact the willingness of the organization towards that particular request. It is not just about the function of leadership or behavior that acquisition leaders need to be aware of, but it is the cultures within individual organizations that drive how a program office responds to the environment and new requirements that needs be understood.   Also, program offices are better equip to handle

difficult situations when they have developed relationships with other program offices within the SoS, and through their contributing influence are able to maintain a high level fitness within the SoS environment.

Recommendations for Future Research

Beyond the scope of this research, some future areas of exploration to study the effects of organizational on an SoS include different architecture implementation within the model, breaking down the resource component between funding, personnel, and talent, moving the selfishness component to global fitness, and finally observing the effect of geographic separation. First, this research explored only a two regular connection architecture (graph), where each system was connected to only its two nearest neighbors. Real world SoS architecture would have systems that vary in the number of connections and not all of them are neighbors. Expansion of this research could be the setting up of a four regular connection, a random selection of connections (Erdos-Renyi graph), and a scale free or a small world network.

Currently, the resource component that is provided by the stimulus or trigger of the model is a generic term that doesn't mean any specific need of a particular system like funding profile, required personnel, or need dates. For example, talent and personnel are not described in the model but are required in order meet and achieve the performance parameters for the system. In DoD acquisition, a required set of people and skills are required in order to meet the objective. By making it generic, the research didn't explore the possibility of improving or degrading a system's performance based on personnel or having the funding necessary to acquire the necessary people, the right skill

and talent levels. In a recent study by Ford, Colburn, and Morris (2012), organizations recognized that experienced individuals are experts with specific skill set. "These individuals can then apply their skill sets to projects with specific Customers, technologies or operational contexts" (Ford, Colburn and Morris 2012). The study also showed that over 90% of interviewed organizations handpick their personnel. To construct a more realistic simulation, future modeling versions should include the appropriate personnel levels for a program office with an allocation for specific skill sets that are desired by leadership.

One of the behavioral scales surprisingly did not correlate with the local fitness as expected. The scale of selfishness was shown to a correlation of 0.106 with local fitness, which indicates that it did not influence the level fitness for a particular. Future research would consider moving this organizational behavior scale to the global fitness calculations. Since the network neighbors provide a level of contribution to a particular system, the system's level of selfishness can determine if its cooperation with other network neighbors based off this scale. If an organization is considered too individualistic, it might reject its neighbors' contribution and its own influence on network neighbors. There is potential in this area to determine how the level of cooperation influences the global fitness within the SoS.

Finally, the model provides no sense of geographic collocation or separation of systems or the improved or degraded performance that accompanies this geographic measure. Research and simulations with these factors could provide an understanding of organizational behavior with additional influence of the positive or negative, based off of the program office's behavior, impact that being collocated or separated has on

performance of the SoS. In addition to this future application, another research thrust is to examine a change of the influences from connected program offices and its ties to geographic location.

Summary

In conclusion, this research provides insight into how acquisition program offices respond to new requirement changes within an SoS. Resources, schedule, and requirements are key inputs into understanding the acquisition process, but organizational behaviors help determine whether not a program office is fit enough to meet the new requirements for a given set of environmental conditions.   Through the examination of Hofstede's cultural dimensions, an organization contains a particular level of its culture's behaviors that could negatively impact a program office's fitness to support a need from the warfighter. Through an acknowledgement and attempting to improve these characteristics, a program office can better meet the needs of the warfighter and their stakeholders even in poor environmental conditions. Finally, program managers need to establish a relationship with other program offices in order to maintain a high level of fitness within the SoS environment.

Appendix A

Trigger Definitions

Upgrade: Known non-critical issues (also critical issues with workarounds) and technological advancement are corrected to the SoS at a known interval of time and provides funding and personnel to Program Office. This is a strategic initiative to improve current systems or replace them, pushed by higher Headquarters and stakeholders.

User Needs: Critical issues (no workaround) and/or deficiencies while operating or testing a system or SoS in field discovered by the warfighter to meet a capability gap. This is supplied by the warfighter or test units to their respective commanders.

Threat: Critical issue that could be exploited by the enemy requiring a technological advancement and/or increase in capability of the SoS. Pushed by think tanks, analysis centers, and intelligence community.

Appendix B

Leadership Index (LI)	Leadership Factor (LF)	Power Distance Index (PDI)	Function(Leadership)

RAND(0,1)	<.5 = 0; >= .5 = 1	TRIA(18,40,62)	IF(PDI < 44 && LF =1), then 1
			IF(PDI >= 44 && LF=0), then 1
			IF(PDI >= 44 && LF=1), then .5
			ELSE =0 (mismatch)

Uncertainty Avoidance Index (UAI)	Ambiguity	Function (Ambiguity)	Individualism/ Collectivism

TRIA(22,46,70)	1-Requirements	IF(UAI>=56 & Ambiguity <=.25), then 1
		IF(UAI>=56 & Ambiguity<=.75), then (rand()*.25)+.25
		IF(UAI>=56 & Ambiguity<=1), then 0
		IF(UAI<56 & Ambiguity <=.25), then 1
		IF(UAI<56 & Ambiguity<=.75), then (rand()*.25)+.50
		IF(UAI<56 & Ambiguity<=1), then (rand()*.25)+.25
			TRIA(66,91,116)

Funct(IND)

IF(IND<=78 && Contextual Influence < .5 && Resources >.5), then rand()*.34
IF(IND<=78 && Contextual Influence < .5 && Resources <.5), then (rand()*.33) + .34
IF(IND<=78 && Contextual Influence > .5 && Resources >.5), then rand()*.34
IF(IND<=78 && Contextual Influence >.5 && Resources <.5), then rand()*.34
IF(IND<=104 && Contextual Influence < .5 && Resources <.5), then (rand()*.33)+.67
IF(IND<=104 && Contextual Influence < .5 && Resources >.5), then (rand()*.33) + .34
IF(IND<=104 && Contextual Influence > .5 && Resources >.5), then rand()*.34
IF(IND<=104 && Contextual Influence >.5 && Resources <.5), then rand()*.33 + .34
IF(IND>104 && Contextual Influence < .5 && Resources <.5), then (rand()*.33)+.67
IF(IND>104 && Contextual Influence < .5 && Resources >.5), then (rand()*.33) + .67
IF(IND>104 && Contextual Influence > .5 && Resources >.5), then (rand()*.33) + .34
IF(IND>104 && Contextual Influence >.5 && Resources <.5), then rand()*.33 + .34

Appendix C

Trigger Set Matrix for M-Selected Systems				
Trigger Type	**Contextual Influence (CI)**	**Resources**	**Schedule**	**Requirements**
IF t < .5, then 1 IF t < .9, then 2 ELSE (>=.9) then 3	For 1: (rand*.50)+.25 For 2: (rand*.25) For 3: (rand*.25)+.75	For 1: (rand*.50)+.25 For 2: (rand*.25)+.75 For 3: (rand*.25)	For 1: (rand*.50)+.25 For 2: (rand*.25) +.75 For 3: (rand*.25)	For 1: (rand*.50)+.25 For 2: (rand*.25) +.75 For 3: (rand*.25)

Appendix D

Willingness, W	1-Selfishness	Ability, A
.5*Funct(Lead)+.5*Funct(Amb)	1- Funct(IND)	3/4*Resources+1/4*Schedule

Local Fitness, F	Weight Up	Weight Down
p*(.4*W+.3*(1-S)+.3*A)+(1-p)*PreviousFitness	rand(0,1)	rand(0,1)

Global Fitness of System, G
LocalFit+wtup*PreviousSysFit+wtdwn*NextSysFit

```matlab
N=1000;
L=10000;
%X = zeros(L, 1);
old = 0;
new = 0;

%ExcelFileName = 'bobsein_12Feb.xlsx';
%delete(ExcelFileName);

%ExcelRow=1;
%header={'N-System','Trigger
Type','PDI','UAI','Ind/Collect','Leadership','Willingness','1-
Selfishness','Ability','LocalFitness','GlobalFitness'};
%xlswrite(ExcelFileName,header,'SoSData','A1');
%ExcelRow =2;

%p is percentage of OB effect vice history (of fitness)
%for p= 0:.2:1
for p=.6
    rand('seed', 11);

%Initialize the System Matrix
System = rand(N,7);
SysAttSet = zeros(N,10);
TriggerSet = zeros(L,5);
StatM = zeros(L,4);
SaveGF = zeros(N,L);
SaveLF = zeros(N,L);
%Initialize the Graph /SoS Architecture
% 2 regular by default
% extensions if time permits for 4 regular, Random Graph, Small World,
% Scale Free; need to change column 8 of System Attribute Matrix to
match
% this set-up

%Establish an initial environment
for x=1
TriggerSet(x,2) = .5;  %Contextual Influence Column 2
TriggerSet(x,3) = 1;  %Resource Column 3
TriggerSet(x,4) = 1;  %Schedule Column 4
TriggerSet(x,5) = 1;  %Requirements Column 5

%System Attribute Matrix = SysAttSet
SysAttSet(:,1) = rand(N,1); %LI = Leadership Index, column 1

for k= 1:N %LeadF = Leadership Factor, column 2
    if SysAttSet(k,1) < 0.5
    SysAttSet(k,2) = 0; %0 =  Transactional Leadership style
    else
    SysAttSet(k,2) = 1; %1 = Transformationsl Leadership style
    end
```

```
end

SysAttSet(:,3) = trirnd(18,40,62,N); %PDI = Power Distance Index,
column 3


for k = 1:N %FunctLead = Function of Leadership and PDI, column 4
    if ((SysAttSet(k,3) < 44) && (SysAttSet(k,2) == 1))
        SysAttSet(k,4) = 1;  %Match of PDI and LeadF (Low PDI and
Transformational)
    elseif ((SysAttSet(k,3) >= 44) && (SysAttSet(k,2) == 0))
        SysAttSet(k,4) = 1;  %Match of PDI and LeadF (High PDI and
Transactional)
    elseif ((SysAttSet(k,3) >= 44) && (SysAttSet(k,2) == 1))
        SysAttSet(k,4) = .5;  %Not a match but transformational tries
to improve this (High PDI and Transactional)
    else
        SysAttSet(k,4) = 0; %Does not match up (Low PDI &
Transactional)
    end
end

SysAttSet(:,5) = trirnd(22,46,70,N); %UAI = Uncertainty Avoidance
Index, column 5


SysAttSet(:,6) = 1-(TriggerSet(x,5)); %Amb = Ambiguity, column 6 in
System Attributes

 %Function of Ambiguity for the System, column 7 in System Attributes
 for k = 1:N
    if ((SysAttSet(k,5) >= 56) && (SysAttSet(k,6) <= .25))
    SysAttSet(k,7) = 1;  %Match of High UAI and Low Ambiguity
    elseif ((SysAttSet(k,5) >= 56) && (SysAttSet(k,6) <= .75))
    SysAttSet(k,7) = (rand*.25)+.25;  %Mis-match of UAI and Ambiguity
    %Due to a High UAI, a program office can attempt the new
requirement
    %but the organizational behavior of the PO hinders this attribute
    else SysAttSet(k,7) = 0;  %Mis-match of UAI and Ambiguity (High
UAI)
    end
    if ((SysAttSet(k,5) < 56) && (SysAttSet(k,6) <= .25))
    SysAttSet(k,7) = 1;  %Match of Low UAI and Low Ambiguity
    elseif ((SysAttSet(k,5) < 56) && (SysAttSet(k,6) <= .75))
    SysAttSet(k,7) = (rand*.25)+.50;
    %Lower UAI implies a better stiuation for program office to handle
    %less requirements than higher UAI POs.
    %Reduction due to less requirements definition
    else SysAttSet(k,7) = (rand*.25)+.25;
    %Lower UAI implies a better stiuation for program office to handle
    %less requirements than higher UAI POs.
    %Reduction due to less requirements definition
    end
 end
SysAttSet(:,8) = 1; %# of Connections of a System, column 8, Since this
is
```

```
%an initial run and it has been pre-determined that each system has 2
%connections then the scaled value for this column is 1.  Any future
experiments
%can randomized the number of connections and set a value per
connection

SysAttSet(:,9) = trirnd(66,91,116,N); %Ind/Col = Individualism vs
Collectivism Index, column 9

%Column 10 Calculations
%Lower Individualism score contributes to a lower selfishness score,
more
%resources and contextual influence can lower that level of selfishness
and
%vice versa for less resources and contextual influence.
%For the purpose of this model, Ind Funct will become the selfishness
%level.
for k = 1:N
    if ((SysAttSet(k,9) <= 78) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)>.5))
    SysAttSet(k,10) = (rand*.34);
    elseif ((SysAttSet(k,9) <= 78) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    elseif ((SysAttSet(k,9) <= 78) && (TriggerSet(x,2)>.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = rand*.34;
    else SysAttSet(k,10) = (rand*.34);
    end
    if ((SysAttSet(k,9) <= 104) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) <= 104) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)>.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    elseif ((SysAttSet(k,9) <= 104) && (TriggerSet(x,2)>.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    else SysAttSet(k,10) = (rand*.34);
    end
    if ((SysAttSet(k,9) > 104) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) > 104) && (TriggerSet(x,2)<.5) &&
(TriggerSet(x,3)>.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) > 104) && (TriggerSet(x,2)>.5) &&
(TriggerSet(x,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    else SysAttSet(k,10) = (rand*.33) + .34;
    end
end
```

```matlab
%Determining Willingness, (1-Selfishness), & Ability for all systems
%                               1st column - willingness, w
%                               2nd column - 1 - selfishness, s
%                               3rd column - ability, a
%                               4th column - local fitness, b
%                               5th column - Delta i-1 => i  (UP)
%                               6th column - Delta i+1 => i  (DOWN)
%                               7th column - Global Fitness of
system, G
for h = 1:N %Willingness (column 1)
    System(h,1) = (.5)*SysAttSet(h,4) + (.5)*SysAttSet(h,7);
end


for h = 1:N %1-sefishness (column 2)
    System(h,2) = 1 - SysAttSet(h,10);
   %the more selfish an organization is, the less fit it becomes
end


for h = 1:N
    System(h,3) = .75*TriggerSet(1,3) + .25*TriggerSet(1,4); %Ability
(column 3)
end


for h = 1:N
   System(h,4) = (p)*(0.4*System(h,1) + 0.3*System(h,2) +
0.3*System(h,3)) + (1-p)*System(h,4);
end


% calculate the global fitness, initial values
System(1,7)=System(1,4)+System(N,4)*System(N,5)+System(2,4)*System(2,6)
;
System(N,7)=System(N,4)+System(N-1,4)*System(N-
1,5)+System(1,4)*System(N,6);
for i = 2:N-1
    System(i,7)=System(i,4)+System(i-1,4)*System(i-
1,5)+System(i+1,4)*System(i+1,6);
end
end
%Initial Values of local and global fitness at each system
System0b = System(:,4);
System0f = System(:,7);

%Write next N blocks to Excel
%SysNum=[1:N]';
%Trig0=zeros(N,1);
%ExcelBlock=[SysNum Trig0 SysAttSet(:,[3 5 9 2]) System(:,[1 2 3 4
7])];
%xlswrite(ExcelFileName,ExcelBlock,'SoSData',strcat('A',num2str(ExcelRo
w)));
%ExcelRow=ExcelRow+N;

for j = 1:L    %Trigger Calculations! ATTRIBUTES of Trigger can be set
here
```

```matlab
t = rand;

if t < .5
        TriggerSet(j,1) = 1;  % trigger 1 is User Need, Column 1
        TriggerSet(j,2) = (rand * .50) + .25;  %Contextual Influence
Column 2
        TriggerSet(j,3) = (rand * .50)+.25;  %Resource Column 3
        TriggerSet(j,4) = (rand * .50)+.25;  %Schedule Column 4
        TriggerSet(j,5) = (rand * .50)+.25;  %Requirements Column 5
elseif t < .9
        TriggerSet(j,1) = 2; % trigger 2 is Upgrade
        TriggerSet(j,2) = (rand * .25);
        TriggerSet(j,3) = (rand * .25) + .75;
        TriggerSet(j,4) = (rand * .25) + .75;
        TriggerSet(j,5) = (rand * .25) + .75;
 else
        TriggerSet(j,1) = 3; % trigger 3 is Threat
        TriggerSet(j,2) = (rand * .25) + .75;
        TriggerSet(j,3) = (rand * .25);
        TriggerSet(j,4) = (rand * .25);
        TriggerSet(j,5) = (rand * .25);
end

% TriggerSet(j,1) = triggertype;
% Trigger Influence ==> Willingness, Selfishness, Ability for set of
nodes (SoS)
%fprintf('Epoch %i - trigger %i\n',j, TriggerSet(j,1))

% create a subset M from the N systems
M = 1:N;
for q = 1:N
    r= randi(N,1);
    temp=M(r);
    M(r)=M(q);
    M(q)=temp;
end
r2=randi(round(.33*N),1);
%************CHANGE  only effect % of nodes!!!!!*******************

% M is a unique, non-redundant subset of N
temp=M(1:r2);
Mprime=M(r2+1:N);
M=temp;

for k = M
   %FunctLead = Function of Leadership and PDI, column 4
    if ((SysAttSet(k,3) < 44) && (SysAttSet(k,2) == 1))
        SysAttSet(k,4) = 1;  %Match of PDI and LeadF (Low PDI and
Transformational)
    elseif ((SysAttSet(k,3) >= 44) && (SysAttSet(k,2) == 0))
        SysAttSet(k,4) = 1;  %Match of PDI and LeadF (High PDI and
Transactional)
    elseif ((SysAttSet(k,3) >= 44) && (SysAttSet(k,2) == 1))
```

```matlab
            SysAttSet(k,4) = .5;   %Not a match but transformational tries
to improve this (High PDI and Transactional)
    else
            SysAttSet(k,4) = 0; %Does not match up (Low PDI &
Transactional)
    end
end


for k = M
SysAttSet(k,6) = 1-(TriggerSet(j,5)); %Amb = Ambiguity, column 6 in
System Attributes
end


for k = M %Function of Ambiguity for the System, column 7 in System
Attributes
    if ((SysAttSet(k,5) >= 56) && (SysAttSet(k,6) <= .25))
    SysAttSet(k,7) = 1;   %Match of High UAI and Low Ambiguity
    elseif ((SysAttSet(k,5) >= 56) && (SysAttSet(k,6) <= .75))
    SysAttSet(k,7) = (rand*.25)+.25;   %Mis-match of UAI and Ambiguity
    %Due to a High UAI, a program office can attempt the new
requirement
    %but the organizational behavior of the PO hinders this attribute
    else SysAttSet(k,7) = 0;   %Mis-match of UAI and Ambiguity (High
UAI)
    end
    if ((SysAttSet(k,5) < 56) && (SysAttSet(k,6) <= .25))
    SysAttSet(k,7) = 1;   %Match of Low UAI and Low Ambiguity
    elseif ((SysAttSet(k,5) < 56) && (SysAttSet(k,6) <= .75))
    SysAttSet(k,7) = (rand*.25)+.50;
    %Lower UAI implies a better stiuation for program office to handle
    %less requirements than higher UAI POs.
    %Reduction due to less requirements definition
    else SysAttSet(k,7) = (rand*.25)+.25;
    %Lower UAI implies a better stiuation for program office to handle
    %less requirements than higher UAI POs.
    %Reduction due to less requirements definition
    end
end


%Column 10 Calculations
%Lower Individualism score contributes to a lower selfishness score,
more
%resources and contextual influence can lower that level of selfishness
and
%vice versa for less resources and contextual influence.
%For the purpose of this model, Ind Funct will become the selfishness
%level.
for k = M
   if ((SysAttSet(k,9) <= 78) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)>.5))
   SysAttSet(k,10) = (rand*.34);
   elseif ((SysAttSet(k,9) <= 78) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)<.5))
   SysAttSet(k,10) = (rand*.33) + .34;
```

```matlab
    elseif ((SysAttSet(k,9) <= 78) && (TriggerSet(j,2)>.5) &&
(TriggerSet(j,3)<.5))
    SysAttSet(k,10) = rand*.34;
    else SysAttSet(k,10) = (rand*.34);
    end
    if ((SysAttSet(k,9) <= 104) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) <= 104) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)>.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    elseif ((SysAttSet(k,9) <= 104) && (TriggerSet(j,2)>.5) &&
(TriggerSet(j,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    else SysAttSet(k,10) = (rand*.34);
    end
    if ((SysAttSet(k,9) > 104) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) > 104) && (TriggerSet(j,2)<.5) &&
(TriggerSet(j,3)>.5))
    SysAttSet(k,10) = (rand*.33) + .67;
    elseif ((SysAttSet(k,9) > 104) && (TriggerSet(j,2)>.5) &&
(TriggerSet(j,3)<.5))
    SysAttSet(k,10) = (rand*.33) + .34;
    else SysAttSet(k,10) = (rand*.33) + .34;
    end
 end


for h = M %Willingness (column 1)
    System(h,1) = (.5)*SysAttSet(h,4)+(.5)*SysAttSet(h,7);
end


for h = M %1-sefishness (column 2), the more selfish an organization
is, the less
    %fit it becomes
     System(h,2) = 1 - SysAttSet(h,10);
end

for h = M
    System(h,3) = .75*TriggerSet(j,3) + .25*TriggerSet(j,4); %Ability
(column 3)
end

for h = M %System Fitness (Local), column 4
 System(h,4) = (p)*(.4*System(h,1) + .3*System(h,2) + .3*System(h,3)) +
(1-p)*System(h,4);
end

%With new influence, recalculate all global fitness for all nodes
%New--Decrease of Local fitness for not being selected
for t = Mprime
    if (TriggerSet(j,1)==1);
    System(t,4) = System(t,4) - System(t,4)*.025;
```

```matlab
    elseif (TriggerSet(j,1)==2);
    System(t,4) = System(t,4) - System(t,4)*.05;
    else
    System(t,4) = System(t,4);
    end
end

System(1,7) =
System(1,4)+System(N,4)*System(N,5)+System(2,4)*System(2,6);
System(N,7) = System(N,4)+System(N-1,4)*System(N-
1,5)+System(1,4)*System(1,6);
for i = 2:N-1
    System(i,7) = System(i,4)+System(i-1,4)*System(i-
1,5)+System(i+1,4)*System(i+1,6);
end

StatM(j,1) = sum((System(:,7))*(1/N));    %Total Value (avg global
fitness) of the SoS at each iteration
StatM(j,2) = TriggerSet(j,1);

%Ability Re-evaluation
for t = Mprime
    System(t,3) = System(t,3) + ((.1*(rand-.5))*System(t,3));
    if System(t,3) > 1
        System(t,3) = 1;
    end
    if System(t,3) < 0
        System(t,3) = 0;
    end
end

LearningMat = zeros(N,11);

%First Column for N-Systems
LearningMat(:,1) = 1:N;

%Column 2 is copying PDI Values from SysAttSet Matrix
LearningMat(:,2) = SysAttSet(:, 3);

%Column 3 is copying UAI Values from SysAttSet Matrix
LearningMat(:,3) = SysAttSet(:, 5);

%Column 4 is copying IND/COL Values from SysAttSet Matrix
LearningMat(:,4) = SysAttSet(:, 9);

%Column 5 is copying Local Fitness Values from System Matrix
LearningMat(:,5) = System (:, 4);

%Column 6 is copying Global Fitness Values from System Matrix
LearningMat(:,6) = System (:, 7);

%Chance to Improve Behavior for M-Selected Systems
for o = M;
```

```matlab
%[x,i] = min(LearningMat(o,6));
LearningMat(o,7) = 1;

if LearningMat(o,7) == 1;
LearningMat(o,8) = rand; %Probability of Learning
end

if LearningMat(o,8) <= .25

    LearningMat(o,9) = trirnd(18,40,62,1); %Revalue the PDI value of
Lowest Fit System, loop
    if (LearningMat(o,9)>= LearningMat(o,2))
        LearningMat(o,9) = LearningMat(o,2);
    else
    end
    if (LearningMat(o,9)<44)
        SysAttSet(o,2) = 1;
    end

    LearningMat(o,10) = trirnd(22,46,70,1); %Revalue the UAI value of
Lowest Fit System
    if (LearningMat(o,10)>= LearningMat(o,3))
        LearningMat(o,10) = LearningMat(o,3);
    else
    end

    LearningMat(o,11) = trirnd(66,96,116,1); %Revalue the IND value of
Lowest Fit System
    if LearningMat(o,11) >= LearningMat(o,4)
        LearningMat(o,11) = LearningMat(o,4);
    else
    end

    SysAttSet(o,3) = LearningMat(o,9);   %Reassigning New Org Beh Values
to System Attribute Matrix
    SysAttSet(o,5) = LearningMat(o,10);
    SysAttSet(o,9) = LearningMat(o,11);

end
end
%StatM(j,3) = LearningMat(o,1);
%StatM(j,4) = LearningMat(o,8);
SaveGF(:,j) = System(:,7);
SaveLF(:,j) = System(:,4);
%write another block to Excelfile

%Trigj=TriggerSet(j,1).*ones(N,1);
%ExcelBlock=[SysNum Trigj SysAttSet(:,[3 5 9 2]) System(:,[1 2 3 4
7])];
%xlswrite(ExcelFileName,ExcelBlock,'SoSData',strcat('A',num2str(ExcelRo
w)));
%ExcelRow=ExcelRow+N;
```

```
end

SystemLb = System(:,4);
SystemLf = System(:,7);

figure
%plot the local and global values... just like the paper
plot(StatM(:,1))
title(strcat('Global SoS Fitness p = ',num2str(p)))
axis([0 L 0 1.5])
%grid

figure
subplot(2,1,1)
[Bnums,Bbins] = hist(System0b, 50);
plot (Bbins,Bnums/N);
axis([0 1 0 .5])
title('Local Fitness B pdf at t=initial')
grid
subplot(2,1,2)
[Bnums,Bbins] = hist(System0f, 50);
plot (Bbins,Bnums/N);
title('Global Fitness F pdf at t=initial')
axis([0 3 0 .5])
grid

figure
subplot(2,1,1)
[Bnums,Bbins] = hist(SystemLb, 20);
plot (Bbins,Bnums/N);
axis([0 1 0 .5])
s=num2str(L);
title(strcat('p=',num2str(p),'% OB,  Local Fitness B pdf at t=',s))
grid
subplot(2,1,2)
[Bnums,Bbins] = hist(SystemLf, 20);
plot (Bbins,Bnums/N);
title(strcat('p=',num2str(p),'% OB,  Global Fitness F pdf at t=',s))
axis([0 3 0 .5])
grid

figure
cm=cumsum(mean(SaveGF))./(1:L);
plot(cm)

end
```

Bibliography

Atkins, Clinton. *CSAF Shares Perspective during AETC Senior Leader Conference.* October 26, 2012. http://www.af.mil/news/story.asp?id=123323730&channelPageId=s6925EC1356510 FB5E044080020E329A9&programId=tA1FBF31D25AE0DF20125FF41AF4B0E56 (accessed November 02, 2012).

Bak, Per, and Kim Sneppen. "Punctuated Equuilibrium and Criticality in a Simple Model of Evolution." *Physical Review Letters*, December 1993: 4083-4086.

Banks, Jerry, John S. Carson II, Barry L. Nelson, and David M. Nicol. *Discrete-Event System Simulation 5th Edition.* Upper Saddle River, New Jersey: Prentice Hall, 2010.

Bartolozzi, M., D. B. Leinweber, and A. W. Thomas. "Symbiosis in the Bak-Sneppen Model for Biological Evolution with Economic Applications." *Physica A: Statistical Mechanics and Its Applications*, June 15, 2006: 499-508.

Borshchev, Andrei, and Alexei Filippov. "From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools." *The 22nd International Conference of the System Dynamics Society.* Oxford, England, 2004.

Chan, Wai Kin Victor, Young-Jun Son, and Charles M. Macal. "Agent-Based Simulation Tutorial - Simulation of Emergent Behavior and Differences between Agent-Based Simulation and Discrete-Event Simulation." *2010 Winter Simulation Conference.* Baltimore, Maryland: IEEE, 2010. 135-150.

Dahmann, Judith, George Rebovich, Ralph Lowery, JoAnn Lane, and Kristen Baldwin. "An Implementers' View of Systems Engineering for Systems of Systems." *Proceedings of IEEE International Systems Conference.* Montreal, 2011.

Delloiacono, Brad. *Remotely Piloted Aircraft (RPA) System of Systems (SoS): Wave Model Application and Analysis.* Thesis, Air Force Institute of Technology, 2012.

DeVellis, Robert F. *Scale Development: Theory and Applications 2nd Edition.* Thousand Oaks, California: Sage Publications, Inc., 2003.

DoD Instruction 5000.02. *Operation of the Defense Acquisition System.* Washington D.C.: Pentagon, 2008.

Ford, Jennifer S., Ryan M. Colburn, and Yosef A. Morris. *Principles of Rapid Acquisition and Systems Engineering.* Air Force Institue of Technology, 2012.

Gersick, Connie J. "Time and Transition in Work Teams: Toward a New Model of Group Development." *Academy of Management Journal* Vol 31, no. 1 (1988): 9-41.

Guralnik, David B. *Webster's New World Dictionary of the American Language, Second College Edition.* United States: William Collins & World Publishing Co., Inc., 1978.

Higham, Desmond J., and Nicholas J. Higham. *MATLAB Guide 2nd Edition.* Philadelphia: Society for Industrial and Applied Mathematics, 2005.

Hofstede, Geert. *Cultures Consequences.* London: Sage Publications, 2001.

Hofstede, Geert. "Motivation, Leadership, and Organization: Do American Theories Apply Aboard?" *Organizational Dynamics*, no. 9 (1980): 42-63.

Hofstede, Geert. "National Cultures in Four Dimensions." *International Studies of management & Organization* Vol 13, no. 1/2 (1983): 46-74.

Macal, Charles M., and Michael J. North. "Tutorial on Agent-Based Modeling and Simulation Part 2: How to Model with Agents." *Proceedings of the 2006 Winter Simulation Conference.* Monterey, California: IEEE, 2006. 73-83.

Maier, Mark W., and Eberhardt Rechtin. *The Art of Systems Architecting.* Boca Raton, FL: CRC Press, 2009.

Middleton, Victor. "Simulating Small Unit Military Operations with Agent-Based Models of Complex Adaptive Systems." *2010 Winter Simulation Conference.* Baltimore, Maryland: IEEE, 2010. 119-134.

OUSD AT&L. *Systems Engineering Guide for Systems of Systems.* Washington D.C.: Pentagon, August 2008.

Pfeffer, Jeffrey, and John F. Veiga. "Putting People Fiirst for Organizational Success." *Academy of Management Executive*, 1999: 37-48.

Reynolds, Craig. "Flocks, Herds, and Schools: A Distributed Behavioral Model." *Computer Graphics*, July 1987: 25-34.

Vecchio, Robert P. *Leadership.* Notre Dame, Indiana: University of Notre Dame Press, 2007.

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)* 21 Mar 2013	2. REPORT TYPE Master's Thesis	3. DATES COVERED *(From – To)* 04 Aug 2011 – 21 Mar 2013
4. TITLE AND SUBTITLE ANALYSIS OF EFFECTS OF ORANIZATIONAL BEHAVIOR ON EVOLVING SYSTEM OF SYSTEMS ACQUISITION PROGRAMS THROUGH AGENT BASED MODELING		**5a. CONTRACT NUMBER**
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Bobsein, Klayton S., Captain, USAF		**5d. PROJECT NUMBER**
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/ENV) 2950 Hobson Way, Building 640 WPAFB OH 45433-8865		**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT-ENV-13-M-03
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Mr. Scott Lucero 703-681-6654 Scott.Lucero@osd.mil DDRE/SE 1901 North Beauregard Street, Suite 500 Alexandria, VA 22311		**10. SPONSOR/MONITOR'S ACRONYM(S)**
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A: APPROVED FOR PUBLICRELEASE; DISTRIBUTION IS UNLIMITED

13. SUPPLEMENTARY NOTES
This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT
Every weapon system operates in context with one or more system of systems (SoS). It is the SoS that provides warfighting capability. Each system is managed independently by a program office with program-centric priorities. These systems must be interconnected and interoperable, so the program office must collaborate across the SoS. The SoS are always changing and evolving, triggered by variety of stimuli. Acquisition program offices can be characterized with a set of organizational behaviors that respond to the environment, are influenced by SoS architecture, and can be described by their fitness and contribution to SoS. Using Geert Hofstede's cultural dimensions, integrated with a modified version of the Bak-Sneppen biological evolutionary model, this research highlights which set of behaviors are significant in affecting the SoS fitness. Through the use of agent based modeling, it was determined that organizational behaviors of willingness and ability were significant factors to local fitness with correlation of 0.548 and 0.535 respectively. Using these factors, a regression model was built to predict the local fitness of the system. Additionally, global fitness was highly dependent on the influence from connected systems. Lastly, the SoS global fitness remained highly stable after a period of time throughout different modeling strategies.

15. SUBJECT TERMS
Agent Based Modeling and Simulation, Organizational Behavior, DoD Acquisitions, System Engineering, Architecture

16. SECURITY CLASSIFICATION OF: Unclassified			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON John M Colombi, Ph.D.
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	107	**19b. TELEPHONE NUMBER** *(Include area code)* (937) 255-6565, x 3347 (john.colombi@afit.edu)

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18